Gulf Journal of Mathematics
Vol 23, Issue 1 (2026) 1-18

ISSN: 2309-4966
https://doi.org/10.56947/ydx7g395

TRIGONOMETRIC B-SPLINE QUASI-INTERPOLANTS AND THEIR APPLICATIONS TO NUMERICAL INTEGRATION AND ELLIPTIC PDES

MANAL NAIMI[1,2]* and MOHAMED LAMNII[2]

Abstract. This work presents the construction and analysis of trigonometric B-splines of orders 2, 3, and 4 for numerical approximation. We develop associated quasi-interpolants and weighted Gaussian quadrature rules, and demonstrate their effectiveness in computing the L2-energy of functions. Additionally, these tools are applied to solve a second-order elliptic differential equation with homogeneous boundary conditions using a classical variational formulation. Numerical experiments confirm the accuracy, stability, and efficiency of the proposed methods.

Keywords. Elliptic PDEs, Gaussian quadrature, Spline, Quasi-interpolation.
2020 Mathematics Subject Classification. Primary 46L55; Secondary 44B20

1. Introduction

For several decades, spline functions have been fundamental tools in numerical analysis and approximation [8]. Their local structure, controlled regularity, and recursive definition make them particularly well suited for the numerical solution of differential equations and the construction of stable and accurate numerical methods. Among the various spline families, trigonometric B-splines are especially effective for modeling periodic and oscillatory phenomena, and have been successfully applied in collocation, quasi-interpolation, and elliptic problems [1, 6].

Recent developments confirm the effectiveness of spline-based methods in both numerical approximation and geometric applications. In particular, spline quasi-interpolants have been used in numerical integration and surface estimation [7], while generalized spline interpolation techniques have been explored in various contexts [9, 5].

The construction of quasi-interpolants based on trigonometric splines relies on Marsden's identity, which provides explicit expressions for reproduction functionals. This property makes quasi-interpolants particularly attractive from both

Date: Received: Mar 10, 2026; Revised: Apr 7, 2026; Accepted: Apr 21, 2026.
* Corresponding author

theoretical and computational perspectives. Moreover, when combined with suitable quadrature rules, these quasi-interpolants lead to efficient numerical integration schemes that exploit the local support of the spline basis. In particular, weighted Gaussian quadrature methods [2] have shown that an appropriate choice of weights significantly enhances integration accuracy.

In this work, we construct explicit trigonometric B-spline quasi-interpolants of orders 2, 3, and 4 using Marsden's identity. Their approximation properties are investigated through numerical experiments, including a numerical convergence study. Furthermore, we develop weighted quadrature rules adapted to the spline basis, which are then applied to two problems: the variational solution of the elliptic equation $-u'' = f$ with homogeneous boundary conditions, and the numerical approximation of the L^2-energy of periodic functions. This approach naturally combines quasi-interpolation with weighted quadrature adapted to the spline basis, providing an effective and coherent numerical framework for both variational problems and the approximation of integral quantities.

The numerical results demonstrate that increasing the spline order significantly improves accuracy, and confirm the effectiveness of combining quasi-interpolation with weighted quadrature for both variational problems and numerical integration.

The remainder of the paper is organized as follows: Section 2 introduces the theoretical background on trigonometric B-splines, Section 3 presents the construction of quasi-interpolants, Section 4 discusses numerical experiments and quadrature rules, and Section 5 presents the applications.

2. Preliminaries

In the context of numerical analysis and approximation theory, L-splines are a natural generalization of classical splines. Given a linear differential operator L, an L-spline is a piecewise-defined function $s(x)$ satisfying

$$L(s(x)) = 0 \quad \text{on each subinterval } [t_i,\, t_{i+1}].$$

A particular class of these, trigonometric B-splines, correspond to L-splines with compact support that form a local basis of the associated spline space. These functions are constructed in such a way as to guarantee restricted support, controlled regularity, and good approximation properties, which makes them particularly well suited to modeling periodic or oscillatory phenomena.

2.1. **Trigonometric B-splines.**

Before introducing trigonometric B-splines of order k, it is necessary to specify the functional space in which they are constructed. To do this, we consider the space of trigonometric polynomials associated with order k, denoted $\mathbb{T}_k$. This space provides the natural basis for defining and analyzing trigonometric splines. The structure of $\mathbb{T}_k$ depends on the parity of k, which leads to the following two definitions:

If k is odd,

$$\mathbb{T}_k = \left\{ f(x) = a_0 + \sum_{n=1}^{\frac{(k-1)}{2}} \left(a_n \cos(nx) + b_n \sin(nx)\right) \quad : \quad a_0,\, a_n,\, b_n \in \mathbb{R} \right\},$$

and if k is even,

$$\mathbb{T}_k = \left\{ f(x) = \sum_{j=0}^{\frac{k}{2}-1} \left(\alpha_j \cos\left(\tfrac{2j+1}{2}x\right) \;+\; \beta_j \sin\left(\tfrac{2j+1}{2}x\right)\right) \quad : \quad \alpha_j,\, \beta_j \in \mathbb{R} \right\}.$$

We now introduce trigonometric B-splines of order k (See [4] and [8]). Let k be a positive integer and $\{t_i\}_{i=1}^{m+k}$ a subdivision of the interval $[a, b]$ satisfying

$$a = t_1 = \ldots = t_k < t_{k+1} < \ldots < t_{m+1} = \ldots = t_{m+k} = b$$

The trigonometric B-spline of order k, denoted $T_i^{(k)}(x)$, is defined recursively as follows.

$$T_{i,1}(x) = \begin{cases} 1, & \text{if } x \in [t_i,\, t_{i+1}), \\ 0, & \text{otherwise.} \end{cases}$$

and

$$T_{i,k}(x) = \frac{\sin\left(\dfrac{x - t_i}{2}\right)}{\sin\left(\dfrac{t_{i+k-1} - t_i}{2}\right)}\, T_{i,k-1}(x) + \frac{\sin\left(\dfrac{t_{i+k} - x}{2}\right)}{\sin\left(\dfrac{t_{i+k} - t_{i+1}}{2}\right)}\, T_{i+1,k-1}(x).$$

Any fraction appearing in the recurrence relation and having a zero denominator is interpreted as being equal to zero.

The derivative of the trigonometric B-spline $T_{i,k}(x)$ is defined by the recurrence relation (See [3])

$$\left(T_{i,k}(x)\right)' \;=\; \left(\frac{k-1}{2}\right) \frac{\cos\left(\frac{x-t_i}{2}\right)}{\sin\left(\frac{t_{i+k}-t_i}{2}\right)}\, T_{i,k-1}(x) \;-\; \frac{\cos\left(\frac{t_{i+k}-x}{2}\right)}{\sin\left(\frac{t_{i+k}-t_i}{2}\right)}\, T_{i+1,k-1}(x).$$

The figures 1, 2, 3 and 4 illustrate trigonometric B-splines of orders 1, 2, 3 and 4.

2.2. **Trigonometric Marsden's Identity.** We aim to represent any function $f \in \mathbb{T}_k$ using a linear combination of trigonometric B-splines. According to [4], the trigonometric version of Marsden's identity is given by

$$\sin^{k-1}\left(\frac{y-x}{2}\right) = \sum_{i=1}^{n} \theta_{k,i}(y)\, T_{i,k}(x), \qquad \text{for all } x, y \in J,$$

where

$$\theta_{k,i}(y) := \prod_{\ell=1}^{k-1} \sin\left(\frac{y - t_{i+\ell}}{2}\right).$$

This identity ensures exact reproduction within the trigonometric spline space $\mathbb{T}_k$.

In order to illustrate the behavior of the trigonometric Marsden's identity, we explicitly give the cases:

for $k = 2$, the trigonometric space is defined by

$$\mathbb{T}_2 = \text{span}\left\{\sin\left(\tfrac{x}{2}\right),\ \cos\left(\tfrac{x}{2}\right)\right\}.$$

FIGURE 1. trigonometric B-spline of order 1.

trigonometric B-spline of order 1.jpg

FIGURE 2. trigonometric B-spline of order 2.

trigonometric B-spline of order 2.jpg

FIGURE 3. trigonometric B-spline of order 3.

trigonometric B-spline of order 3.jpg

FIGURE 4. trigonometric B-spline of order 4.

trigonometric B-spline of order 4.jpg

and the trigonometric Marsden's identity are given by

$$\cos\left(\tfrac{x}{2}\right) = \sum_{i=1}^{m} \cos\left(\tfrac{t_{i+1}}{2}\right)\ T_{i,2}(x),$$

$$\sin\left(\tfrac{x}{2}\right) = \sum_{i=1}^{m} \sin\left(\tfrac{t_{i+1}}{2}\right)\ T_{i,2}(x).$$

To illustrate this reproduction property, the figures 5 and 6 show the error between each exact function and its reconstruction via Marsden's identity.

FIGURE 5. Reproduction error of the function $\cos\left(\frac{x}{2}\right)$ obtained by Marsden's identity.

error for cos(x-2) for order 2.jpg

FIGURE 6. Reproduction error of the function $\sin\left(\frac{x}{2}\right)$ obtained by Marsden's identity.

error for sin(x-2) for order 2.jpg

for $k = 3$, the trigonometric space is defined by

$$\mathbb{T}_3 = \text{span}\left\{ 1,\ \sin(x),\ \cos(x) \right\}.$$

and the trigonometric Marsden's identity are given by

$$\begin{aligned} 1 &= \sum_{i=1}^{m} \cos\left(\frac{t_{i+1} - t_{i+2}}{2}\right) T_{i,3}(x), \\ \cos(x) &= \sum_{i=1}^{m} \cos\left(\frac{t_{i+1} + t_{i+2}}{2}\right) T_{i,3}(x), \\ \sin(x) &= \sum_{i=1}^{m} \sin\left(\frac{t_{i+1} + t_{i+2}}{2}\right) T_{i,3}(x). \end{aligned}$$

The figures 7, 8 and 9 show the error between each exact function and its reconstruction via Marsden's identity.

For $k = 4$, the trigonometric space is defined by

$$\mathbb{T}_4 = \text{span}\left\{\sin\left(\tfrac{x}{2}\right),\ \cos\left(\tfrac{x}{2}\right),\ \sin\left(\tfrac{3x}{2}\right),\ \cos\left(\tfrac{3x}{2}\right)\right\}.$$

and the trigonometric Marsden's identity are given by

$$\cos\left(\tfrac{x}{2}\right) = \sum_{i=1}^{m} \frac{1}{3}\left[2\cos\left(\tfrac{t_{i+1}}{2}\right)\cos\left(\tfrac{t_{i+2}-t_{i+3}}{2}\right) + \cos\left(\tfrac{t_{i+1}-t_{i+2}-t_{i+3}}{2}\right)\right] T_{i,4}(x),$$

$$\sin\left(\tfrac{x}{2}\right) = \sum_{i=1}^{m} \frac{1}{3}\left[2\sin\left(\tfrac{t_{i+1}}{2}\right)\cos\left(\tfrac{t_{i+2}-t_{i+3}}{2}\right) - \sin\left(\tfrac{t_{i+1}-t_{i+2}-t_{i+3}}{2}\right)\right] T_{i,4}(x),$$

FIGURE 7. Reproduction error of the constant function 1 obtained by Marsden's identity.

error for 1 for order 3.jpg

FIGURE 8. Reproduction error of the function $\cos(x)$ obtained by Marsden's identity.

error for cos(x) for order 3.jpg

FIGURE 9. Reproduction error of the function $\sin(x)$ obtained by Marsden's identity.

error for sin(x) for order 3.jpg

$$\cos\left(\tfrac{3x}{2}\right) = \sum_{i=1}^{m} \cos\left(\tfrac{t_{i+1}+t_{i+2}+t_{i+3}}{2}\right)\, T_{i,4}(x),$$

$$\sin\left(\tfrac{3x}{2}\right) = \sum_{i=1}^{m} \sin\left(\tfrac{t_{i+1}+t_{i+2}+t_{i+3}}{2}\right)\, T_{i,4}(x).$$

The figures 10,11, 12 and 13 show the error between each exact function and its reconstruction via Marsden's identity.

FIGURE 10. Reproduction error of the function $\cos\left(\frac{x}{2}\right)$ obtained by Marsden's identity.

error for cos(x-2) for order 4.jpg

FIGURE 11. Reproduction error of the function $\sin\left(\frac{x}{2}\right)$ obtained by Marsden's identity.

error for sin(x-2) for order 4.jpg

FIGURE 12. Reproduction error of the function $\cos\left(\frac{3x}{2}\right)$ obtained by Marsden's identity.

error for cos(3x-2) for order 4.jpg

FIGURE 13. Reproduction error of the function $\sin\left(\frac{3x}{2}\right)$ obtained by Marsden's identity.

error for sin(3x-2) for order 4.jpg

3. TRIGONOMETRIC QUASI-INTERPOLANT

The construction of the trigonometric quasi-interpolant is based directly on Marsden's identity presented in the previous section. This identity allows each basis function of the space $\mathbb{T}_k$ to be expressed as an explicit combination of trigonometric B-splines of order k, providing the associated reproduction coefficients . The idea is then to exploit these coefficients as evaluation functionals applied to the function to be approximated. This process leads to a simple linear operator that reproduces exactly all the functions of $\mathbb{T}_k$. In the following, we introduce the quasi-interpolants thus obtained for orders $k = 2, 3$, and 4. All numerical results reported in this section are computed using a uniform discretization with mesh size $h = \frac{2\pi}{n}$, with $n = 500$.

The trigonometric quasi-interpolant of order 2 is given by

$$Q_2 f(x) = \sum_{i=1}^{m} f(t_{i+1}) T_{i,2}(x).$$

The Table 1 shows the maximum error made by the trigonometric quasi-interpolant of order 2 for several test functions. We observe that the first three functions, which belong to the space $\mathbb{T}_2$, are reproduced with machine precision, confirming the exact reproduction property of the quasi-interpolant. For functions outside $\mathbb{T}_2$, the error is no longer zero but remains relatively small, confirming the approximation capability of the quasi-interpolant even outside its reproduction space. These results illustrate both the theoretical fidelity of the quasi-interpolant in $\mathbb{T}_2$ and its numerical efficiency on more general functions.

Table 1. Maximum error of the quasi-interpolant of order 2 for different test functions.

Case	Exact function $f(x)$	Error $\max \lvert f(x) - Q_2 f(x) \rvert$
1	$\cos\left(\frac{x}{2}\right)$	2.2204×10^{-16}
2	$\sin\left(\frac{x}{2}\right)$	3.3307×10^{-16}
3	$0.4\cos\left(\frac{x}{2}\right) + 0.8\sin\left(\frac{x}{2}\right)$	3.3307×10^{-16}
4	$\cos(x)$	1.4804×10^{-5}
5	$\sin(x)$	1.4702×10^{-5}
6	$-6\cos^2(x)\sin(x) + 3\sin^3(x)$	3.9689×10^{-4}
7	$\exp(\sin x)$	3.9961×10^{-5}

Now, for $k = 3$ and for $\alpha = \frac{2\pi}{n}$ with $n \in \mathbb{N}$, the trigonometric Quasi-interpolant is given by

$$Q_3 f(x) = \sum_{i=1}^{m} \lambda_i(f)\, T_{i,3}(x),$$

where the functional are given by

$$\lambda_i(f) = \frac{1}{4}\sec\left(\frac{\alpha}{2}\right)\big(-f((i-3)\alpha) + f((i-1)\alpha) + 2(\cos(\alpha)+1)\, f((i-2)\alpha)\big), \qquad i = 3, \dots, m-2,$$

and

$$\lambda_1(f) = f(0),$$

$$\lambda_2(f) = \frac{1}{4}\sec\left(\frac{\alpha}{2}\right)\big(-f(2\alpha) + 2(\cos(\alpha)+1)\, f(\alpha) + f(0)\big),$$

$$\lambda_{m-1}(f) = \frac{1}{4}\sec\left(\frac{\alpha}{2}\right)\left(-f(2\pi - 2\alpha) + 2(\cos(\alpha)+1)\, f(2\pi - \alpha) + f(2\pi)\right),$$
$$\lambda_m(f) = f(2\pi).$$

TABLE 2. Maximum error of the quasi-interpolant of order 3 for different test functions.

Case	Exact function $f(x)$	Error $\max_x \lvert f(x) - Q_3 f(x)\rvert$
1	1	4.4409×10^{-16}
2	$\cos(x)$	9.9920×10^{-16}
3	$\sin(x)$	8.8471×10^{-16}
4	$\cos\left(\frac{x}{2}\right)$	9.8836×10^{-8}
5	$\sin\left(\frac{x}{2}\right)$	9.8762×10^{-8}
6	$-6\cos(x)^2\sin(x) + 3\sin(x)^3$	1.4250×10^{-5}
7	$\exp(\sin x)$	7.3603×10^{-7}

The results in Table 2 confirm the exact reproduction of the functions belonging to the $\mathbb{T}_3$ space, namely 1, $\cos(x)$, and $\sin(x)$, for which the maximum error remains of the order of machine precision, as was observed for order 2. We also observe that several functions that are not part of either $\mathbb{T}_2$ or $\mathbb{T}_3$ are correctly approximated by both quasi-interpolants. However, the numerical results show that the order 3 quasi-interpolant systematically gives a smaller error than the order 2 quasi-interpolant. This improvement can be explained by the fact that the space $\mathbb{T}_3$ is richer than $\mathbb{T}_2$, which allows it to better represent the variations of the test functions and obtain a more accurate approximation.

After establishing the quasi-interpolants for $k = 2$ and $k = 3$, we now consider the case $k = 4$, the trigonometric quasi-interpolant is given by

$$Q_4 f(x) = \sum_{i=1}^{m} \lambda_{4,i}(f)\, T_{i,4}(x),$$

where

$$\lambda_{4,i}(f) = \frac{1}{6}\sec\left(\tfrac{\alpha}{2}\right)\left[-f((i-3)\alpha) + 8\cos^3\left(\tfrac{\alpha}{2}\right) f((i-2)\alpha) - f((i-1)\alpha)\right], \qquad i = 3, \dots, m-2,$$

and

$$\lambda_{4,1}(f) = f(0),$$

$$\lambda_{4,2}(f) = \frac{1}{12\cos(\alpha)+6}\Big[(4\cos(\alpha)+3)\sec\left(\tfrac{\alpha}{2}\right) f(0) + 2(2\cos(\alpha)+1)^2 f(\alpha) - (2\cos(\alpha)+1)^2\sec\left(\tfrac{\alpha}{2}\right) f(2\alpha) + 2f(3\alpha)\Big],$$

$$\lambda_{4,m-1}(f) = \frac{1}{12\cos(\alpha)+6}\Big[(4\cos(\alpha)+3)\sec\left(\tfrac{\alpha}{2}\right) f(2\pi) + 2(2\cos(\alpha)+1)^2 f(2\pi - \alpha)$$

$$-(2\cos(\alpha)+1)^2\sec\left(\tfrac{\alpha}{2}\right)f(2\pi-2\alpha)+2f(2\pi-3\alpha)\Big],$$

$$\lambda_{4,m}(f)=f(2\pi).$$

TABLE 3. Maximum error of the quasi-interpolant of order 4 for different test functions.

Case	Exact function $f(x)$	Error $\max_x \lvert f(x)-Q_4f(x)\rvert$
1	$\cos\left(\frac{x}{2}\right)$	6.6613×10^{-16}
2	$\sin\left(\frac{x}{2}\right)$	6.6613×10^{-16}
3	$\cos\left(\frac{3x}{2}\right)$	2.7756×10^{-15}
4	$\sin\left(\frac{3x}{2}\right)$	2.6090×10^{-15}
5	$\cos(x)$	9.6201×10^{-10}
6	$\sin(x)$	7.0945×10^{-10}
7	$-6\cos^2(x)\sin(x)+3\sin^3(x)$	1.0092×10^{-7}
8	$\exp(\sin x)$	4.2412×10^{-9}

The results of the Table 3 clearly confirm the advantage of order 4. Compared to quasi-interpolants of orders 2 and 3, order 4 provides significantly lower errors , both for functions belonging to $\mathbb{T}_4$ and for those outside this space. This improvement can be explained by the increased richness of $\mathbb{T}_4$, which allows for better capture of the variations in the test functions. Thus, the quasi-interpolant of order 4 appears to be the most performant of the three approaches studied.

Numerical convergence of the quasi-interpolants. While the previous results illustrate the reproduction properties and approximation behavior of the quasi-interpolants, we now study their numerical convergence. To this end, we consider several smooth 2π-periodic test functions that are not exactly reproduced by the quasi-interpolants.

For each quasi-interpolant, the approximation error is measured in the maximum norm over $[0,2\pi]$, namely

$$E_n(f)=\max_{x\in[0,2\pi]}|f(x)-Q_kf(x)|,$$

where $k=2,3,4$.

The experimental order of convergence is computed using the general formula

$$\text{order}=\frac{\ln(E_n/E_m)}{\ln(m/n)}.$$

The computations are performed for successive refinements of the partition, with

$$h=\frac{2\pi}{n},$$

and for the following test functions:

$$f_1(x) = e^{\sin x}, \qquad f_2(x) = \frac{1}{2+\cos x}, \qquad f_3(x) = \sin(2x) + \cos(3x).$$

Tables 4–6 report the errors and the corresponding experimental orders of convergence for the quasi-interpolants Q_2, Q_3, and Q_4, respectively.

TABLE 4. Maximum errors and experimental orders of convergence for the quasi-interpolant Q_2.

n	$E_n(f_1)$	order	$E_n(f_2)$	order	$E_n(f_3)$	order
20	2.3464×10^{-2}	—	8.1595×10^{-3}	—	1.4220×10^{-1}	—
50	4.0190×10^{-3}	1.93	1.4506×10^{-3}	1.89	2.3438×10^{-2}	1.96
100	1.0030×10^{-3}	2.00	3.6787×10^{-4}	1.99	5.9335×10^{-3}	2.00
200	2.5107×10^{-4}	2.00	9.2410×10^{-5}	2.00	1.4875×10^{-3}	2.00
500	3.9961×10^{-5}	2.00	1.4392×10^{-5}	2.00	2.3762×10^{-4}	2.00

TABLE 5. Maximum errors and experimental orders of convergence for the quasi-interpolant Q_3.

n	$E_n(f_1)$	order	$E_n(f_2)$	order	$E_n(f_3)$	order
20	1.0668×10^{-2}	—	5.2461×10^{-3}	—	1.0905×10^{-1}	—
50	7.2756×10^{-4}	2.95	3.7411×10^{-4}	2.91	7.8011×10^{-3}	2.98
100	9.2076×10^{-5}	2.98	4.7634×10^{-5}	2.97	9.8461×10^{-4}	2.99
200	1.1537×10^{-5}	3.00	5.9755×10^{-6}	2.99	1.2357×10^{-4}	3.00
500	7.3603×10^{-7}	3.00	3.8277×10^{-7}	3.00	7.9171×10^{-6}	3.00

TABLE 6. Maximum errors and experimental orders of convergence for the quasi-interpolant Q_4.

n	$E_n(f_1)$	order	$E_n(f_2)$	order	$E_n(f_3)$	order
20	1.3756×10^{-3}	—	1.0565×10^{-3}	—	1.7543×10^{-2}	—
50	4.1802×10^{-5}	3.96	3.5730×10^{-5}	3.92	5.5951×10^{-4}	3.98
100	2.6307×10^{-6}	3.99	2.3522×10^{-6}	3.98	3.7787×10^{-5}	3.99
200	1.6552×10^{-7}	4.00	1.4910×10^{-7}	4.00	2.3956×10^{-6}	4.00
500	4.2412×10^{-9}	4.00	3.8150×10^{-9}	4.00	6.0752×10^{-8}	4.00

The numerical results presented in Tables 4–6 clearly illustrate the convergence behavior of the proposed quasi-interpolants.

For the quasi-interpolant Q_2, the experimental orders of convergence are close to 2 for all test functions. Although slightly lower values are observed for coarse discretizations, the convergence order stabilizes as n increases, indicating a second-order behavior.

Similarly, Table 5 shows that the quasi-interpolant Q_3 exhibits experimental convergence orders close to 3, suggesting a stable third-order approximation.

Finally, Table 6 shows that the quasi-interpolant Q_4 achieves experimental orders close to 4 across all refinements, indicating a fourth-order behavior.

Moreover, the consistency of the results across the different test functions highlights the robustness of the proposed quasi-interpolants. In all cases, the approximation error decreases significantly as the mesh is refined.

In addition, for a fixed discretization level, higher-order quasi-interpolants yield significantly smaller errors, highlighting the advantage of increasing the spline order. These observations are in agreement with the expected approximation behavior of higher-order spline quasi-interpolants.

4. Resolution of $-u''(x) = f(x)$

In this section, we illustrate the use of fourth-order trigonometric B-splines in the context of the variational solution of a simple elliptic differential equation. To do this, we consider a one-dimensional problem with homogeneous boundary conditions, which is a classic test case for evaluating the accuracy of spline approximation methods. The objective is to approximate the exact solution from a weak formulation and to analyze the numerical behavior of the method when the solution function has different regularities.

We consider the following boundary value problem:

$$\begin{cases} -u''(x) = f(x), & x \in (0, 2\pi), \\ u(0) = u(2\pi) = 0. \end{cases}$$

By adopting the classical variational formulation, we search for a function $u \in H_0^1(0, 2\pi)$ such that

$$\int_0^{2\pi} u'(x)\, v'(x)\, dx = \int_0^{2\pi} f(x)\, v(x)\, dx, \qquad \forall v \in H_0^1(0, 2\pi).$$

We approximate $u(x)$ as a linear combination of trigonometric B-splines:

$$u_h(x) = \sum_{j=1}^{m} c_j\, T_{j,4}(x),$$

and the test functions are chosen from the same basis, $v(x) = T_{i,4}(x)$. Substituting into the variational formulation, we obtain

$$\sum_{j=1}^{m} c_j \int_0^{2\pi} T'_{j,4}(x)\, T'_{i,4}(x)\, dx = \int_0^{2\pi} f(x)\, T_{i,4}(x)\, dx, \qquad i = 1, \dots, m.$$

Since we impose $u(0) = u(2\pi) = 0$, we get:

$$0 = u(0) = \sum_{j=1}^{m} c_j\, T_{j,4}(0) = c_1\, T_{1,4}(0) = c_1,$$

$$0 = u(2\pi) = \sum_{j=1}^{m} c_j\, T_{j,4}(2\pi) = c_m\, T_{m,4}(2\pi) = c_m.$$

Then, we obtain

$$\sum_{j=2}^{m-1} c_j \int_0^{2\pi} T'_{j,4}(x)\, T'_{i,4}(x)\, dx = \int_0^{2\pi} f(x)\, T_{i,4}(x)\, dx, \qquad \forall\, i = 2, \ldots, m-1.$$

This leads to the matrix system $Kc = F$, where

$$K_{ij} = \int_0^{2\pi} T'_{j,4}(x)\, T'_{i,4}(x)\, dx, \qquad F_i = \int_0^{2\pi} f(x)\, T_{i,4}(x)\, dx.$$

We decompose the derivative $T'_{i,4}(x)$ as

$$T'_{i,4}(x) = \frac{3\cos\left(\frac{x-t_i}{2}\right)}{2\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{i,3}(x) \;-\; \frac{3\cos\left(\frac{t_{i+4}-x}{2}\right)}{2\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{i+1,3}(x).$$

then,

$$\begin{aligned} K_{ij} &= \int_0^{2\pi} \frac{9\cos\left(\frac{x-t_j}{2}\right)\cos\left(\frac{x-t_i}{2}\right)}{4\sin\left(\frac{t_{j+4}-t_j}{2}\right)\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{j,3}(x)\, T_{i,3}(x)\, dx \\ &\quad - \int_0^{2\pi} \frac{9\cos\left(\frac{x-t_j}{2}\right)\cos\left(\frac{t_{i+4}-x}{2}\right)}{4\sin\left(\frac{t_{j+4}-t_j}{2}\right)\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{j,3}(x)\, T_{i+1,3}(x)\, dx \\ &\quad - \int_0^{2\pi} \frac{9\cos\left(\frac{t_{j+4}-x}{2}\right)\cos\left(\frac{x-t_i}{2}\right)}{4\sin\left(\frac{t_{j+4}-t_j}{2}\right)\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{j+1,3}(x)\, T_{i,3}(x)\, dx \\ &\quad + \int_0^{2\pi} \frac{9\cos\left(\frac{t_{j+4}-x}{2}\right)\cos\left(\frac{t_{i+4}-x}{2}\right)}{4\sin\left(\frac{t_{j+4}-t_j}{2}\right)\sin\left(\frac{t_{i+4}-t_i}{2}\right)}\, T_{j+1,3}(x)\, T_{i+1,3}(x)\, dx. \end{aligned}$$

To calculate these integrals, we used weighted Gaussian quadrature. Since the coefficients K_{ij} involve terms of the form $T_{j,3}(x)\, T_{i,3}(x)$, the quadrature rule was constructed using this product as the weight function.

This approach allows us to take into account the local support of the trigonometric splines and obtain an accurate approximation of the integrals. Thus, for any regular function g, the weighted integral is approximated by

$$\int_a^b g(x)\, T_{i,3}(x)\, T_{j,3}(x)\, dx \;\approx\; \sum_{\ell=0}^{3} w_\ell^{(i,j)}\, g\left(x_\ell^{(i,j)}\right),$$

where $\{x_\ell^{(i,j)}\}$ and $\{w_\ell^{(i,j)}\}$ denote respectively the nodes and weights of the quadrature adapted to the weight $W_{i,j}(x) = T_{i,3}(x)\, T_{j,3}(x)$. The quadrature rules are constructed locally on each interval by taking into account the support of the trigonometric B-spline basis functions. In this setting, the weights are naturally adapted to the spline functions, which act as weight functions in the integration process.

This local construction allows the quadrature to accurately capture the behavior of the basis functions, leading to improved numerical integration, especially in the presence of oscillatory or localized features.

Similarly, the second member is evaluated using a weighted Gaussian quadrature adapted to the spline support $T_i(x)$. Thus,

$$F_i = \int_0^{2\pi} f(x)\, T_i(x)\, dx \;\approx\; w_0^{(i)} f\left(x_0^{(i)}\right) + w_1^{(i)} f\left(x_1^{(i)}\right) + w_2^{(i)} f\left(x_2^{(i)}\right) + w_3^{(i)} f\left(x_3^{(i)}\right),$$

where $\{x_\ell^{(i)}\}_{\ell=0}^3$ and $\{w_\ell^{(i)}\}_{\ell=0}^3$ denote respectively the nodes and weights of the quadrature associated with the weight $T_i(x)$.

By combining this approximation with the one used for the coefficients K_{ij}, we finally obtain the linear system

$$Kc = F.$$

Thanks to the compact support of trigonometric B-splines, the stiffness matrix K is sparse and banded, which allows us to directly obtain

$$c = K^{-1}F.$$

Numerical Results. To illustrate the numerical performance of the proposed method, we consider several test functions for which the exact solution is known. This allows us to compare the approximate solution obtained using trigonometric B-splines with the analytical solution.

For each test case, the right-hand side $f(x)$ is computed from the equation

$$-u''(x) = f(x).$$

The approximation error is measured in the maximum norm over $[0, 2\pi]$, namely

$$E_n = \max_{x\in[0,2\pi]} |u(x) - u_h(x)|,$$

where u_h denotes the numerical solution.

We consider the following test functions:

$$u_1(x) = (\cos x - 1)^3 \sin^2(x), \qquad u_2(x) = (1 - \cos x)^2, \qquad u_3(x) = \sin^4(x).$$

The computations are performed for different values of n, with mesh size $h = \frac{2\pi}{n}$. Table 7 reports the corresponding errors for the considered test functions.

Table 7. Errors and experimental orders of convergence for the numerical solution for different values of n.

n	Error (u_1)	order	Error (u_2)	order	Error (u_3)	order
20	2.1345×10^{-3}	—	4.2484×10^{-2}	—	2.1589×10^{-1}	—
50	5.7777×10^{-5}	3.94	1.3657×10^{-3}	3.75	5.1811×10^{-3}	4.07
100	3.6426×10^{-6}	3.99	9.0129×10^{-5}	3.92	3.5582×10^{-4}	3.87
200	2.2092×10^{-7}	4.04	5.7649×10^{-6}	3.97	2.2985×10^{-5}	3.95
500	4.7997×10^{-9}	4.18	1.4948×10^{-7}	3.99	5.9762×10^{-7}	3.98

The results clearly show that the error decreases as the mesh is refined for all test functions, confirming the convergence of the proposed method. The convergence behavior is consistent across the different test cases, demonstrating the robustness of the numerical approach. The observed convergence orders are close to 4, indicating a fourth-order behavior of the proposed numerical method.

Figures 14–16 illustrate the comparison between the exact solutions and their numerical approximations for the considered test cases. The numerical and exact solutions are almost indistinguishable, which confirms the high accuracy of the proposed method and is consistent with the error values reported in Table 7.

FIGURE 14. Exact solution and approximate solution for $u(x) = (\cos x - 1)^3 \sin^2(x)$

equation-(cos(x)-1)^3sin(x)^2.jpg

FIGURE 15. Exact solution and approximate solution for $u(x) = (1 - \cos x)^2$

equation-(1-cos(x))^2.jpg

FIGURE 16. Exact solution and approximate solution for $u(x) = \sin^4(x)$

equation-sin(x)^4.jpg

5. Energy Calculation

In many contexts in numerical analysis, the L^2 norm is an essential tool. It allows us to naturally quantify the amplitude or overall energy of a function defined on a given domain.

In this application, we consider the L^2 energy of a function f, defined by

$$\|f\|_{L^2}^2 = \int_0^{2\pi} f(x)^2\, dx.$$

In order to obtain a numerical approximation of this quantity, we first apply the quasi-interpolant $Q_4 f(x)$. We then obtain

$$\|f\|_{L^2}^2 \;\approx\; \int_0^{2\pi} (Q_4 f(x))^2\, dx = \int_0^{2\pi} \left(\sum_{i=1}^{m} \lambda_i(f)\, T_{i,4}(x) \right)^2 dx.$$

Expanding this square, we get

$$\int_0^{2\pi} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i(f)\, \lambda_j(f)\, T_{i,4}(x)\, T_{j,4}(x)\, dx = \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i(f)\, \lambda_j(f) \int_0^{2\pi} T_{i,4}(x)\, T_{j,4}(x)\, dx.$$

This expression can be rewritten in compact matrix form as

$$\|f\|_{L^2}^2 \;\approx\; c^T M\, c,$$

where

$$c = \begin{pmatrix} \lambda_1(f) \\ \lambda_2(f) \\ \vdots \\ \lambda_m(f) \end{pmatrix}, \qquad M_{ij} = \int_0^{2\pi} T_{i,4}(x)\, T_{j,4}(x)\, dx.$$

To numerically evaluate the coefficients of the matrix M,

$$M_{ij} = \int_0^{2\pi} T_{i,4}(x)\, T_{j,4}(x)\, dx,$$

we use the fourth-order Gauss quadrature rule defined above, taking $f(x) = 1$ and considering the product $T_{i,4}(x)\, T_{j,4}(x)$ as the weight function.

This approach provides an accurate and consistent approximation of the mass matrix associated with the trigonometric spline basis.

Numerical Results. To evaluate the accuracy of the proposed method for the numerical approximation of the L^2-energy, we consider several 2π-periodic test functions.

The error is defined as the absolute difference between the exact energy and its numerical approximation.

We consider the following test functions:

$$f_1(x) = \cos(x), \qquad f_2(x) = \sin(3x), \qquad f_3(x) = (1 - \cos x)^2.$$

The computations are performed for different values of n, with mesh size $h = \frac{2\pi}{n}$.

The experimental order of convergence is computed between two successive discretizations. Table 8 reports the corresponding errors and observed orders.

Table 8. Errors and experimental orders of convergence in the numerical approximation of the L^2-energy for different values of n.

n	Error (f_1)	order	Error (f_2)	order	Error (f_3)	order
20	1.0283×10^{-3}	—	8.0339×10^{-2}	—	7.0212×10^{-4}	—
50	3.5702×10^{-5}	4.15	2.6246×10^{-3}	4.01	1.9087×10^{-5}	4.00
100	2.4509×10^{-6}	3.86	1.6778×10^{-4}	3.97	1.2020×10^{-6}	3.99
200	1.6022×10^{-7}	3.94	1.0531×10^{-5}	3.99	7.5263×10^{-8}	4.00
500	4.2110×10^{-9}	3.98	2.6986×10^{-7}	4.00	1.9277×10^{-9}	4.00

The results clearly show that the error decreases as the mesh is refined for all test functions, confirming the accuracy of the proposed quadrature method. The convergence behavior is consistent across the different cases, including more oscillatory functions such as $\sin(3x)$. The observed convergence orders are close to 4, which confirms the high accuracy of the proposed quadrature-based approximation of the L^2-energy.

6. Conclusion

This work shows that trigonometric B-splines provide an effective tool for numerical approximation. Using Marsden's identity, we constructed simple quasi-interpolants based on trigonometric B-splines of orders 2, 3, and 4. The numerical tests indicate that the approximation improves as the order of the spline increases.

These quasi-interpolants were also used to construct weighted quadrature formulas. The obtained results show that this approach provides accurate approximations for weighted integrals and for the numerical solution of elliptic problems such as the equation $-u'' = f$ and the approximation of the L^2 energy.

These results illustrate the usefulness of trigonometric spline quasi-interpolants for numerical integration and approximation problems. Future work may extend this approach to higher-order splines or to two-dimensional problems.

References

1. Bouhiri, S., Lamnii, A., & Lamnii, M. (2018). Cubic quasi-interpolation spline collocation method for solving convection–diffusion equations. *Mathematics and Computers in Simulation*, *164*, 33–45. https://doi.org/10.1016/j.matcom.2018.11.003
2. Laurie, D. P. (2001). Computation of Gauss-type quadrature formulas. In P.-J. Laurent, P. Sablonnière, & L. L. Schumaker (Eds.), *Curve and surface fitting: Saint-Malo 1999* (pp. 313–329). Vanderbilt University Press. https://doi.org/10.1016/b978-0-444-50617-7.50014-8
3. Lyche, T., & Winther, R. (1979). A stable recurrence relation for trigonometric B-splines. *Journal of Approximation Theory*, *25*(3), 266–279. https://doi.org/10.1016/0021-9045(79)90017-0
4. Lyche, T., Schumaker, L. L., & Stanley, S. (1998). Quasi-interpolants based on trigonometric splines. *Journal of Approximation Theory*, *95*(2), 280–309. https://doi.org/10.1006/jath.1998.3196

5. Mouhoub, C., Mohamed-Yassir Nour, & Mohamed-Yassir Nour. (2026). A new cubic generalized Hermite spline interpolation. Gulf Journal of Mathematics, 22(1), 1-12. https://doi.org/10.56947/gjom.v22i1.3819
6. Naimi, M., & Lamnii, M. (2024). Solving the Laplace equation on the disc using the UAT spline. *Mathematics and Computers in Simulation*. Advance online publication. https://doi.org/10.1016/j.matcom.2024.09.004
7. Saidi, S., Abdelaziz Abbadi, & Mohamed Tahrichi. (2024). Estimating parametric surface areas using spline quasi-interpolants. Gulf Journal of Mathematics, 18(1), 54-71. Retrieved from https://gjom.org/index.php/gjom/article/view/2370
8. Schumaker, L. L. (2007). *Spline functions: Basic theory* (3rd ed.). Cambridge University Press. https://doi.org/10.1017/cbo9780511618994
9. Sefti, R., Sbibih, D., & Jennane, R. (2026). Fine-tuned cubic generalized composite spline interpolation with optimal parameter. Gulf Journal of Mathematics, 22(1), 1-22. https://doi.org/10.56947/gjom.v22i1.3945

[1] Research center, School of Higher Engineering studies, EHEI, Oujda, Morocco.

Email address: naimi.manala@gmail.com

[2] Laboratory LCSI2A, Faculty of Sciences, University Mohamed First, Oujda, Morocco.

Email address: m$_l$amnii1@yahoo.fr