Gulf Journal of Mathematics
Vol 21, Issue 2 (2025) 259-274

ISSN: 2309-4966
https://doi.org/10.56947/gjom.v21i2.3622

A NEW QUADRILATERAL FINITE ELEMENT BASED ON QUADRATIC SPLINE

FATIMA OUMELLAL[1], CHAIMAE MOUHOUB[2], MOHAMED LAMNII[2*] and ABDELLAH LAMNII[3]

Abstract. This study presents a novel quadrilateral finite element constructed from quadratic cardinal B-splines to solve two-dimensional partial differential equations. The formulation is developed through a tensor-product approach, first defined in one dimension and then generalized to two dimensions. Its effectiveness is examined using two representative test cases: an elliptic Dirichlet problem and a convection–diffusion equation with constant coefficients, both discretized on quadrilateral meshes. The proposed element demonstrates superior smoothness, enhanced local approximation accuracy, and improved numerical stability compared with traditional biquadratic Lagrange elements.

Keywords. Finite element, Interpolation, Spline, Convection-diffusion equation.
2020 Mathematics Subject Classification. Primary 65N30, 46L55, 65D07, 41A15, 35K20; Secondary 44B20, 65M60, 41A05, 47A53, 35J05.

1. Introduction

Quadrilateral meshes are widely used in finite element methods due to their ability to provide accurate and efficient approximations over structured domains. When combined with carefully designed interpolation schemes, these meshes enable precise representation of both geometry and solution spaces, making them particularly suitable for solving partial differential equations (PDEs) in various scientific and engineering applications.

Two-dimensional convection-diffusion problems pose significant numerical challenges, especially when convective effects dominate. In such cases, solutions may develop thin boundary layers or sharp gradients, requiring numerical methods that can capture these features without introducing instability. Finite element methods provide the necessary flexibility to discretize complex domains while maintaining stability and convergence.

Traditional finite elements, such as biquadratic Lagrange elements, offer a simple interpolation framework with reasonable accuracy for smooth solutions [20, 9].

Date: Received: Sep 28, 2025; Revised: Oct 26, 2025 Accepted: Nov 7, 2025.
* Corresponding author
© The Author(s) 2025. This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of the licence, visit https://creativecommons.org/licenses/by-nc-nd/4.0/.

However, they can be limited by reduced smoothness across element boundaries, suboptimal conditioning of the system matrices, and decreased performance when approximating steep gradients or non-uniform meshes [4]. Spline-based finite elements, particularly those constructed from B-splines, provide a natural alternative: they inherently preserve higher-order continuity, improve numerical conditioning, and often achieve superior approximation quality with fewer degrees of freedom [6, 2].

Several studies have explored spline-based finite element constructions. Abbadi et al. [1] investigated quasi-interpolants derived from the B-form of B-splines. Lamnii et al. [10, 11] studied B-spline representations for quadratic Sibson-Thomson splines and the computation of Hermite interpolation using polar forms. Further contributions by Lamnii et al. [12, 14] focused on normalized bases for $\mathcal{C}^1$ cubic superspline spaces on Powell-Sabin triangulations, as well as local quasi-interpolants based on multivariate quadratic spline spaces over refined quadrangulations. Speleers [18, 19] proposed normalized B-splines for smooth spline spaces on Powell-Sabin triangulations and introduced multivariate normalized Powell-Sabin B-splines along with quasi-interpolants. More recently, Toshniwal introduced quadratic splines on quadrilateral-triangle meshes with a non-nested refinement scheme that preserves spline definitions along domain boundaries and demonstrates excellent approximation properties.

Although tensor-product constructions are classical in finite element formulations, the novelty of the present work lies in the explicit use of quadratic cardinal B-splines to generate quasi-Lagrange basis functions. This approach combines the local interpolation property of Lagrange polynomials with the smoothness, improved conditioning, and enhanced approximation quality inherent to spline-based representations, leading to a balanced and efficient formulation for convection–diffusion problems.

Building on these developments, this work proposes a new quadrilateral finite element based on quadratic cardinal B-splines. The element is defined using four strategically chosen univariate interpolation points $-1, -1/3, 1/3, 1$ to optimize both numerical stability and interpolation accuracy. This construction maintains the smoothness and convergence properties of B-splines while offering distinct advantages over standard biquadratic Lagrange elements: better conditioning of the stiffness matrices, improved approximation in regions with steep gradients, and enhanced adaptability to non-uniform meshes.

The proposed element is applied to two-dimensional convection-diffusion problems with Dirichlet boundary conditions. Numerical experiments demonstrate optimal convergence rates of $O(h^3)$ in the L^2-norm and $O(h^2)$ in the H^1-norm. A comparative study with radial basis function (RBF) methods is also included, detailing kernel selection, shape parameters, and node distributions. The results confirm that the B-spline-based element provides a robust, accurate, and computationally efficient alternative to classical Lagrange formulations.

The paper is organized as follows. Section 1 presents the univariate properties of quadratic B-splines. Section 2 extends these concepts to quadrilateral finite elements using tensor-product constructions and analyzes their approximation

properties. Section 3 provides numerical examples for convection-diffusion problems and compares the proposed approach with existing methods. The conclusion discusses potential extensions to non-uniform meshes and more complex PDEs.

2. Interpolation by quadratic splines and applications

2.1. **Definition and properties of cardinal B-spline functions.** In applied mathematics, various quadratic finite elements have been studied for interpolation and numerical integration. This work aims to reduce the polynomial space required to construct such elements while preserving accuracy. We introduce a new quadratic spline interpolant with compact support, enabling efficient approximation and quadrature formulas over arbitrary intervals $[a, b]$. The proposed approach maintains desirable convergence properties while optimizing computational cost, making it suitable for practical implementations in numerical analysis and scientific computing.

Definition 2.1. A B-spline is typically defined as a piecewise polynomial function formed by a linear combination of basis splines with minimal compact support. The present study investigates the cardinal quadratic B-spline, defined recursively as follows:

$$B_0(x) = \begin{cases} 1, & \text{si } x \in [0,1] \\ 0, & \text{otherwise} \end{cases}$$

$$B_m(x) = (B_{m-1} * B_0)(x) = \int_{\mathbb{R}} B_{m-1}(x-t)\, B_0(t)\, dt = \int_0^1 B_{m-1}(x-t)\, dt$$

Proposition 2.2. *We state here some essential properties of the cardinal B-spline functions:*

(1) *The support of $B_m(x)$ is the closed interval $[0, m]$.*
(2) *The cardinal B-splines of degrees m and $m-1$ are related by the recurrence formula:*

$$B_m(x) = \frac{x}{m} B_{m-1}(x) + \frac{m-(x-1)}{m} B_{m-1}(x-1), \quad m \geq 1.$$

(3) *The derivative of $B_m(x)$ is given by:*

$$B'_m(x) = B_{m-1}(x) - B_{m-1}(x-1), \quad m \geq 1.$$

[7]

Using the recursive definition, the quadratic cardinal B-spline $B_2(x)$, along with its first derivative $B'_2(x)$, can be expressed in a piecewise form as follows:

Corollary 2.3. *The quadratic B-spline $B_2(x)$ is given by:*

$$B_2(x) = \begin{cases} \frac{1}{2}x^2, & 0 \leq x < 1, \\ -x^2 + 3x - \frac{3}{2}, & 1 \leq x < 2, \\ \frac{1}{2}(3-x)^2, & 2 \leq x < 3, \\ 0, & \textit{otherwise.} \end{cases}$$

Its first derivative is:

$$B_2'(x) = \begin{cases} x, & 0 \le x < 1, \\ 3 - 2x, & 1 \le x < 2, \\ x - 3, & 2 \le x < 3, \\ 0, & \textit{otherwise.} \end{cases}$$

[7]

Figure 1 illustrates the shape of the quadratic cardinal B-spline $B_2(x)$ and its first derivative $B_2'(x)$.

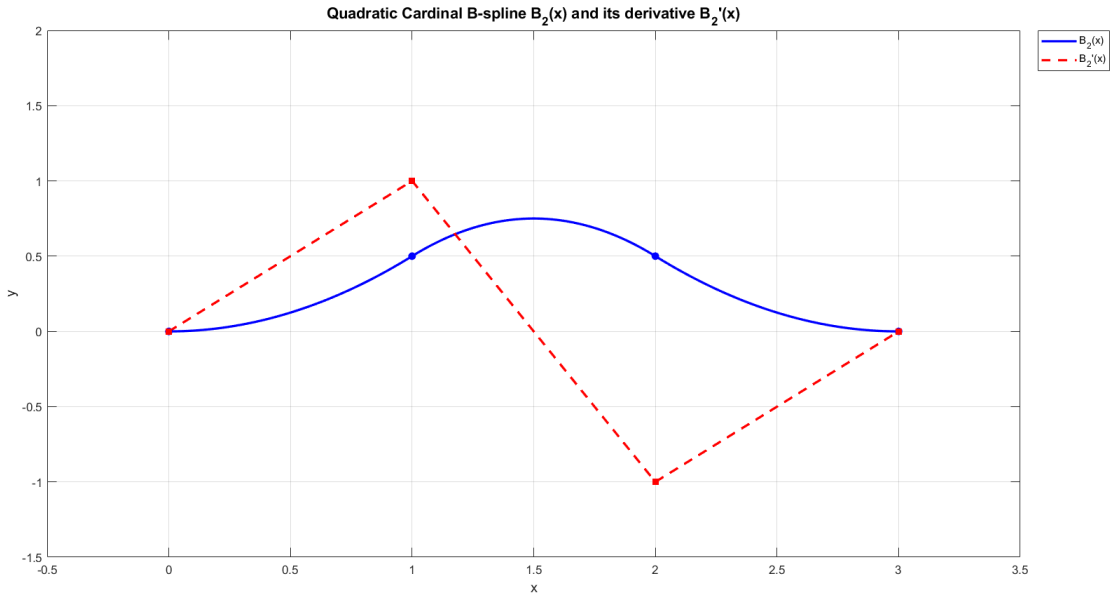


FIGURE 1. The quadratic cardinal B-spline $B_2(x)$ and its derivative $B_2'(x)$.

2.2. **Interpolation on the Interval** $[-1, 1]$**.** The objective of this section is to reduce the polynomial space used in the construction of such elements. To this end, we introduce a novel quadratic spline interpolant that enables the construction of both an interpolation and a quadrature formula on any interval $[a, b]$.

Consider the interval $I = [-1, 1]$ equipped with the uniform partition $\mathcal{X} = \{-1, 0, 1\}$. Let $\mathbf{P}_2$ We denote by $\mathbb{P}_2$ the space of quadratic polynomials, and define $\mathcal{S}(I, \mathcal{X})$ as the span of the shifts $B_2(x + i)$ for $i = 0, \ldots, 3$.

Here, we address the Lagrange interpolation problem defined as follows:

Problem 2.4. Find a quadratic spline $s \in \mathcal{S}(I, \mathcal{X})$ such that

$$s(x_j) = f_j, \quad j = 1, \ldots, 4,$$

where $x_1 = -1, \quad x_2 = -\frac{1}{3}, \quad x_3 = \frac{1}{3}, \quad x_4 = 1$ are the interpolation points, and f_j, $j = 1, \ldots, 4$, are the prescribed data values.
Note that this interpolation problem cannot be solved within the classical quadratic polynomial space $\mathbf{P}_2$ alone.

The interpolation problem admits a unique solution if the Schoenberg–Whitney conditions are satisfied. More precisely, the i-th interpolation point x_i must lie within the support of the corresponding i-th B-spline basis function. In our framework, this condition is readily verified since each x_i belongs to the support of $B_2(x + i)$ for $0 \leq i \leq 3$.

A natural approach to solving this interpolation problem consists in expressing the spline function s in the B-spline basis as

$$s(x) = \sum_{j=0}^{3} c_j B_2(x + j),$$

where the coefficients $\{c_j\}_{j=0}^{3}$ are obtained by enforcing the interpolation conditions.

Using straightforward linear algebra, the coefficients c_j, for $j = 0, \ldots, 3$, are explicitly given by

$$\begin{aligned} c_0 &= -\frac{1}{16}f_0 + \frac{9}{16}f_1 - \frac{27}{16}f_2 + \frac{35}{16}f_3, \quad c_1 = \frac{1}{16}f_0 - \frac{9}{16}f_1 + \frac{27}{16}f_2 - \frac{3}{16}f_3, \\ c_2 &= -\frac{3}{16}f_0 + \frac{27}{16}f_1 - \frac{9}{16}f_2 + \frac{1}{16}f_3, \quad c_3 = \frac{35}{16}f_0 - \frac{27}{16}f_1 + \frac{9}{16}f_2 - \frac{1}{16}f_3. \end{aligned}$$

To isolate the values f_j and simplify their use in the subsequent formulation, we reorganize the interpolation expression by grouping together the B-splines that are multiplied by the same coefficients. This leads to the so-called quasi-Lagrange form:

$$s(x) = \sum_{j=1}^{4} f_j L_j(x), \tag{2.1}$$

where every quasi-Lagrange function L_j can be represented as a finite linear combination of integer-translated copies of the B-spline $B_2(x)$.
For clarity, we provide below the explicit forms of these functions:

$$\begin{aligned} L_1(x) &= -\tfrac{1}{16}\big(B_2(x) - B_2(x+1) + 3B_2(x+2) - 35B_2(x+3)\big), \\ L_2(x) &= +\tfrac{9}{16}\big(B_2(x) - B_2(x+1) + 3B_2(x+2) - 3B_2(x+3)\big), \\ L_3(x) &= -\tfrac{9}{16}\big(3B_2(x) - 3B_2(x+1) + B_2(x+2) - B_2(x+3)\big), \\ L_4(x) &= +\tfrac{1}{16}\big(35B_2(x) - 3B_2(x+1) + B_2(x+2) - B_2(x+3)\big). \end{aligned}$$

The figure below displays the graphs of the spline functions L_j, for $j = 1, \cdots, 4$. Each of these functions belongs to the class $\mathcal{C}^1(\mathbb{R})$, meaning they are continuously differentiable on the real line. Moreover, the symmetry relations $L_4(-x) = L_1(x)$ and $L_3(-x) = L_2(x)$ hold, as a direct consequence of the even symmetry of the quadratic B-spline $B_2(x)$ with respect to its midpoint. These relations highlight the inherent mirror symmetry of the quasi-Lagrange basis functions.

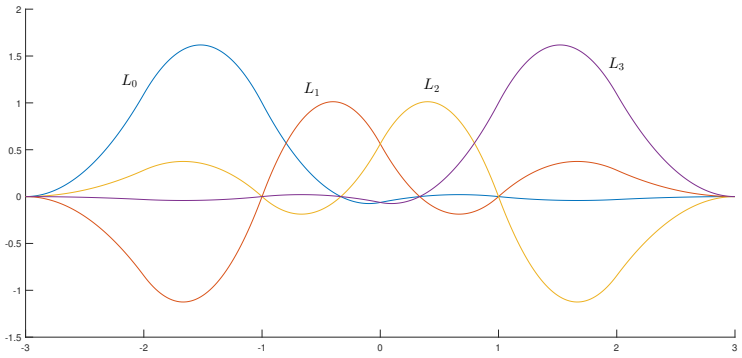


FIGURE 2. The graphs of the spline functions $L_i, i = 1, \ldots, 4$.

2.3. **Interpolation on an Arbitrary Interval $[a, b]$.** In this subsection, we adapt the interpolant defined by equation (2.1) to approximate a function on an arbitrary interval. We provide the expression of the interpolation error between the adapted interpolant and the target function.In addition, we construct a quadrature rule by integrating the $\mathcal{C}^0$ quadratic spline interpolant $\mathcal{I}f$ over the interval $[a, b]$, and we include a numerical example to demonstrate the performance of this approach.

To this end, we first introduce the following notations and preliminary definitions.

Consider a closed and bounded interval $[a, b] \subset \mathbb{R}$, and let $\tau = \{x_0, x_1, \ldots, x_n\}$ denote a partition of $[a, b]$, where

$$a = x_0 < x_1 < \cdots < x_n = b.$$

For each subinterval $[x_i, x_{i+1}]$, define the midpoint

$$x_{i+\frac{1}{2}} := \frac{x_i + x_{i+1}}{2}.$$

We consider the spline space

$$S(\tau, [a, b]) = \left\{ u \in C^0([a, b]) : u|_{[x_i, x_{i+1}]} \in S_{\tau_i}, \quad i = 0, \ldots, n-1 \right\},$$

where

$$S_{\tau_i} = \left\{ u \in C^1([x_i, x_{i+1}]) : u|_{[x_i, x_{i+\frac{1}{2}}]}, u|_{[x_{i+\frac{1}{2}}, x_{i+1}]} \in \mathbb{P}_2 \right\}.$$

That is, on each subinterval, the function u is a piecewise quadratic spline, continuous on $[a, b]$ and continuously differentiable on each sub-subinterval defined by the midpoint.

For convenience, we define four interpolation points on each subinterval $[x_i, x_{i+1}]$ as

$$x_{i,1} = x_i, \quad x_{i,2} = \frac{2x_i + x_{i+1}}{3}, \quad x_{i,3} = \frac{x_i + 2x_{i+1}}{3}, \quad x_{i,4} = x_{i+1}.$$

Set

$$h_i := x_{i+1} - x_i,$$

and define the affine mapping

$$\theta_i(x) := \frac{2}{h_i}\left(x - x_{i+\frac{1}{2}}\right),$$

which maps the interval $[x_i, x_{i+1}]$ onto $[-1, 1]$.

2.3.1. *Construction of the Interpolant.* Given values $f_{i,j}$, $i = 0, \ldots, n-1$, $j = 1, \ldots, 4$, a unique spline $u \in S(\tau, [a, b])$ can be found such that

$$u(x_{i,j}) = f_{i,j}, \quad \text{for all } i, j.$$

Within every subinterval $[x_i, x_{i+1}]$, the spline can be expressed as

$$u(x) = \sum_{j=1}^{4} f_{i,j} L_j(\theta_i(x)),$$

where $\{L_j\}_{j=1}^4$ denote the local Lagrange basis functions corresponding to the points $\theta_i(x_{i,j})$ within the interval $[-1, 1]$.

2.3.2. *Interpolation Error.* The error between the spline interpolant u and a sufficiently smooth function f can be bounded as follows.

Proposition 2.5. *Suppose $f \in C^3([a, b])$. Then, for all $x \in [a, b]$,*

$$|f(x) - u(x)| \le C \, \|f^{(3)}\|_\infty h^3,$$

where $h = \max_{0 \le i \le n-1} h_i$, and

$$C = \frac{S\,M}{48},$$

with S and M independent of f and h.

Proof. On each subinterval $[x_i, x_{i+1}]$, the spline interpolant u is defined by

$$u(x) = \sum_{j=1}^{4} f(x_{i,j})\, L_j(\theta_i(x)), \quad \text{where } \theta_i(x) = \frac{2(x - x_i)}{h_i} - 1 \in [-1, 1].$$

The local basis functions $\{L_j\}_{j=1}^4$ (previously defined) satisfy the interpolation conditions

$$L_j(x_{i,k}) = \delta_{jk}, \qquad j, k = 1, \ldots, 4,$$

at the normalized nodes $\{-1, -\frac{1}{3}, \frac{1}{3}, 1\}$.

Let $f \in C^3([a, b])$ and denote by P its quadratic Taylor polynomial around x:

$$f(t) = P(t) + R(t), \qquad R(t) = \frac{f^{(3)}(\xi_t)}{6}(t - x)^3, \quad \xi_t \in [a, b].$$

Since the spline space $S(\tau, [a, b])$ is exact for all quadratic polynomials,

$$u_P(x) = P(x),$$

and thus the interpolation error depends only on the remainder:

$$f(x) - u(x) = R(x) - u_R(x).$$

Using the interpolation property $u_R(x_{i,j}) = R(x_{i,j})$, one obtains

$$R(x) - u_R(x) = \sum_{j=1}^{4} \Big(R(x) - R(x_{i,j}) \Big) L_j(\theta_i(x)).$$

Substituting $R(x_{i,j}) = \frac{f^{(3)}(\xi_{i,j})}{6}(x_{i,j} - x)^3$ and $|x_{i,j} - x| \leq \frac{h_i}{2}$, we get

$$|f(x) - u(x)| \leq \frac{\|f^{(3)}\|_\infty}{6} \left(\frac{h_i}{2}\right)^3 \sum_{j=1}^{4} |L_j(\theta_i(x))| \, |\theta_i(x)^3 - \theta_j^3|.$$

Setting

$$S = \max_{\theta \in [-1,1]} \sum_{j=1}^{4} |L_j(\theta)|, \qquad M = \max_{j, \theta \in [-1,1]} |\theta^3 - \theta_j^3|,$$

we obtain the uniform bound

$$|f(x) - u(x)| \leq \frac{S\,M}{48} \, \|f^{(3)}\|_\infty \, h^3.$$

For the normalized nodes $\{-1, -\frac{1}{3}, \frac{1}{3}, 1\}$, a numerical evaluation gives

$$S = \frac{20563}{5000}, \qquad M = 2,$$

hence

$$C = \frac{S\,M}{48} = \frac{\frac{20563}{5000} \times 2}{48} = \frac{20563}{120000}.$$

Therefore,

$$|f(x) - u(x)| \leq \frac{20563}{120000} \, \|f^{(3)}\|_\infty \, h^3.$$

This completes the proof. □

2.3.3. *Quadrature Formula.* By integrating the spline interpolant $\mathcal{I}f := u$ over each subinterval, we obtain the quadrature formula

$$\int_{x_i}^{x_{i+1}} \mathcal{I}f(x)\, dx = \frac{h_i}{8} \left(f_{i,1} + 3f_{i,2} + 3f_{i,3} + f_{i,4} \right).$$

Hence, the composite quadrature formula for approximating $\int_a^b f(x)\, dx$ is

$$\int_a^b f(x)\, dx \approx \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} \mathcal{I}f(x)\, dx = \frac{1}{8} \sum_{i=0}^{n-1} h_i \left(f_{i,1} + 3f_{i,2} + 3f_{i,3} + f_{i,4} \right).$$

If the partition is uniform, i.e., $h_i = h = \frac{b-a}{n}$, and $f \in C^3([a,b])$, the quadrature error satisfies

$$\left| \int_a^b f(x)\, dx - \int_a^b \mathcal{I}f(x)\, dx \right| \leq C(b-a) \|f^{(3)}\|_\infty h^3,$$

with the same constant C as above.

2.3.4. *Numerical Example.* We test the interpolation and quadrature formulas on the function

$$f_1(x) = e^x \sin(2\pi x), \quad x \in [-1, 1].$$

Consider the uniform partition

$$x_i = -1 + ih, \quad i = 0, \ldots, 2^k,$$

with step size $h = 1/2^{k-1}$ for $k \in \mathbb{N}$.

For $k = 2, \ldots, 7$, define the interpolation error

$$E(f, k) := \max_{0 \le j \le n} |\mathcal{I}f(x_j) - f(x_j)|,$$

where $n = 2^{k+5}$ is the number of evaluation points (including points between interpolation nodes to test accuracy). The numerical convergence order (NCO) is computed as

$$\mathrm{NCO}(k) := \log_2 \left(\frac{E(f, k)}{E(f, k-1)} \right), \quad k = 3, \ldots, 7.$$

k	$E(f, k)$	NCO(k)
2	1.0523×10^{-1}	–
3	1.2350×10^{-2}	3.09
4	1.7812×10^{-3}	2.79
5	2.3562×10^{-4}	2.92
6	3.0227×10^{-5}	2.96
7	3.8261×10^{-6}	2.98

Table 1 summarizes the maximum interpolation errors and the observed convergence orders, indicating third-order accuracy, consistent with the theoretical estimate.

This confirms the effectiveness of the quadratic spline interpolant and the corresponding quadrature formula for smooth functions on arbitrary intervals.

3. Quadrilateral Finite Element

In this part, we present the formulation and main characteristics of quadrilateral finite elements. We first establish a reference square, then explain how to map this reference element onto a general quadrilateral within the domain. This approach allows the construction of interpolation functions through the tensor product of one-dimensional basis functions, and it facilitates the analysis of their approximation capabilities on quadrilateral meshes.

Let $\Omega \subset \mathbf{R}^2$ be a bounded domain with a piecewise smooth boundary $\partial\Omega$, and let $\mathcal{T}_h$ denote a convex, non-degenerate quadrilateral mesh of Ω.

We introduce $\widehat{K}$ as the reference square $[-1, 1] \times [-1, 1]$ and denote by K an arbitrary quadrilateral element within the mesh $\mathcal{T}_h$.

Suppose that the vertices of $\widehat{K}$ are given by

$$s_1 := (-1, -1), \quad s_2 := (1, -1), \quad s_3 := (1, 1), \quad s_4 := (-1, 1).$$

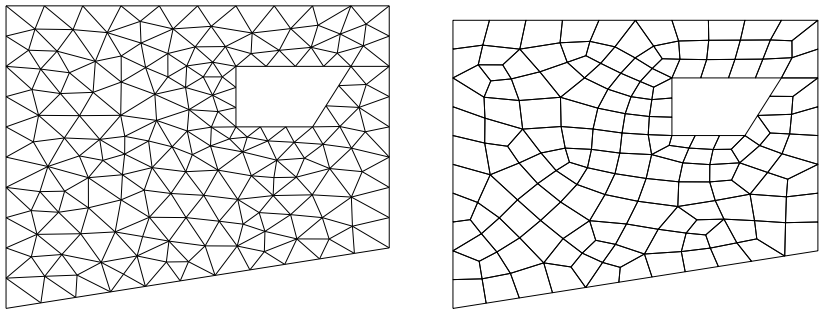


FIGURE 3. On the left, a triangulation of Ω; on the right, a quadrangulation obtained by recombining the triangulation on the left.

The vertices of K are denoted by

$$V_1 := (x_1, y_1), \quad V_2 := (x_2, y_2), \quad V_3 := (x_3, y_3), \quad V_4 := (x_4, y_4).$$

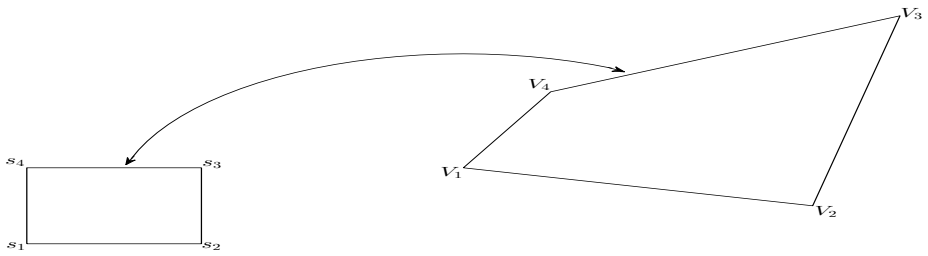


FIGURE 4. Transformation from the reference element $\widehat{K}$ to the generic quadrilateral element K. Here, we first consider the case where K is a rectangle.

Let $\left(V_{ij}^x, V_{ij}^y\right)$ denote the Cartesian coordinates of the node V_{ij}, and let f be a function of two variables defined on the rectangle K. Using the tensor product of the one-dimensional interpolation given by the basis functions L_i, we define the two-variable interpolant $\mathcal{I}f(x, y)$ on K by:

$$\mathcal{I}f(x,y) = \sum_{i=1}^{4}\sum_{j=1}^{4} f_{ij}\left(V_{ij}^x, V_{ij}^y\right) L_i(\theta(x))L_j(\tilde{\theta}(y)),$$

where $\theta(x)$ and $\tilde{\theta}(y)$ are suitable transformations mapping the coordinates x and y onto the reference domain.

The interpolant $\mathcal{I}f$ defined by the above formula is exact on $\mathcal{P}_2 \otimes \mathcal{P}_2$, which means it reproduces all monomials $x^i y^j$ for $i, j = 0, 1, 2$. In particular, the coordinate functions satisfy

$$x = \sum_{i=1}^{4}\sum_{j=1}^{4} V_{ij}^x L_i(\theta(x))L_j(\tilde{\theta}(y)), \quad y = \sum_{i=1}^{4}\sum_{j=1}^{4} V_{ij}^y L_i(\theta(x))L_j(\tilde{\theta}(y)).$$

By introducing the variables $\xi = \theta(x)$ and $\eta = \tilde{\theta}(y)$, we consider the transformation

$$x = \sum_{i=1}^{4}\sum_{j=1}^{4} V_{ij}^{x} L_i(\xi) L_j(\eta), \quad y = \sum_{i=1}^{4}\sum_{j=1}^{4} V_{ij}^{y} L_i(\xi) L_j(\eta).$$

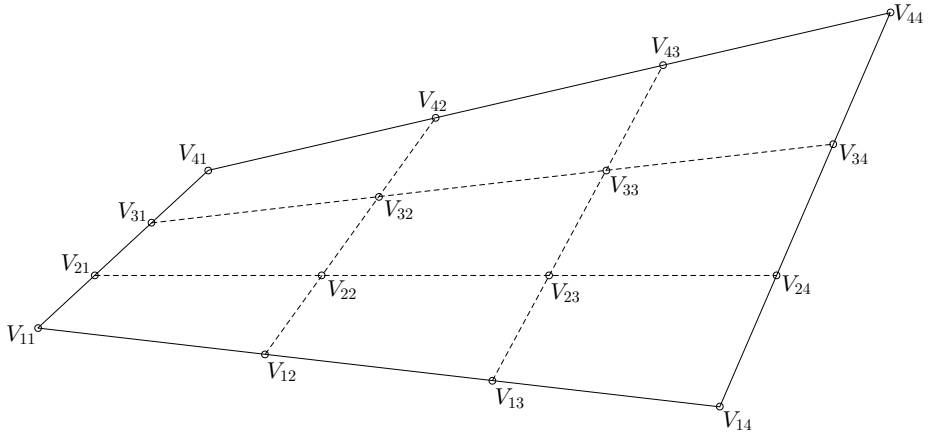


FIGURE 5. Mapping from the reference square to the physical element via the basis functions L_i.

Let us denote the corner nodes as follows:

$$V_{11} = V_1, \quad V_{14} = V_2, \quad V_{44} = V_3, \quad V_{41} = V_4.$$

The other nodes V_{ij} are expressed as weighted averages of these vertices:

$$\begin{aligned} V_{12} &= \tfrac{1}{3}(2V_1 + V_2), \quad V_{13} = \tfrac{1}{3}(V_1 + 2V_2), \\ V_{21} &= \tfrac{1}{3}(2V_1 + V_4), \quad V_{31} = \tfrac{1}{3}(V_1 + 2V_4), \\ V_{24} &= \tfrac{1}{3}(2V_2 + V_3), \quad V_{34} = \tfrac{1}{3}(V_2 + 2V_3), \\ V_{42} &= \tfrac{1}{3}(2V_4 + V_3), \quad V_{43} = \tfrac{1}{3}(V_4 + 2V_3), \end{aligned}$$

and for the interior nodes,

$$\begin{aligned} V_{32} &= \tfrac{1}{9}(2V_1 + 4V_4 + V_2 + 2V_3), \\ V_{33} &= \tfrac{1}{9}(V_1 + 2V_4 + 2V_2 + 4V_3), \\ V_{22} &= \tfrac{1}{9}(4V_1 + 2V_4 + 2V_2 + V_3), \\ V_{23} &= \tfrac{1}{9}(2V_1 + V_4 + 4V_2 + 2V_3). \end{aligned}$$

4. Numerical Application

4.1. **Poisson Problem.** Consider the Poisson equation on a polygonal domain $\Omega \subset \mathbb{R}^2$:

$$-\Delta u = f \quad \text{in } \Omega, \qquad u = 0 \quad \text{on } \Gamma_D \subset \partial\Omega,$$

with $f \in L^2(\Omega)$. Its variational formulation reads: find $u \in H_0^1(\Omega)$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, dx\, dy = \int_\Omega f v \, dx\, dy, \quad \forall v \in H_0^1(\Omega).$$

The finite element method relies on a quadrangulation Q_h of Ω and defines a discrete space $V_{0h} \subset H_0^1(\Omega)$ of continuous piecewise functions vanishing on Γ_D. The approximate solution $u_h \in V_{0h}$ satisfies

$$\int_\Omega \nabla u_h \cdot \nabla v_h \, dx\, dy = \int_\Omega f v_h \, dx\, dy, \quad \forall v_h \in V_{0h}.$$

Expressing u_h in the nodal basis $\{\phi_i\}_{i=1}^N$,

$$u_h(x,y) = \sum_{i=1}^{N} U_i \phi_i(x,y),$$

the global linear system is

$$AU = b,$$

with

$$A_{ij} = \int_\Omega \nabla \phi_i \cdot \nabla \phi_j \, dx\, dy, \qquad b_i = \int_\Omega f \phi_i \, dx\, dy.$$

On each quadrilateral element K, the physical coordinates (x,y) are obtained via an isoparametric mapping from the reference square $[-1,1]^2$:

$$x(\xi,\eta) = \sum_{p,q} V_{pq}^x L_p(\xi) L_q(\eta), \quad y(\xi,\eta) = \sum_{p,q} V_{pq}^y L_p(\xi) L_q(\eta),$$

where L_p, L_q are the quasi-Lagrangian basis functions associated with the reference element nodes.

The elemental stiffness matrix is defined by

$$A_{pq}^K = \int_{-1}^{1}\int_{-1}^{1} \left(\nabla_{\xi,\eta} L_p \cdot J_K^{-1} J_K^{-\top} \nabla_{\xi,\eta} L_q\right) \det(J_K)\, d\xi\, d\eta,$$

and the elemental mass matrix by

$$M_{pq}^K = \int_{-1}^{1}\int_{-1}^{1} L_p(\xi,\eta) L_q(\xi,\eta) \det(J_K)\, d\xi\, d\eta,$$

where J_K is the Jacobian matrix of the isoparametric mapping. All integrals are evaluated numerically using the chosen quadrature rule, ensuring the consistency and accuracy of the scheme.

4.1.1. *Numerical Results.* To validate our approach, consider the Poisson problem on the unit square:

$$-\Delta u = f \quad \text{in } \Omega = (0,1)^2, \qquad u = 0 \quad \text{on } \partial\Omega,$$

whose exact solution is

$$u(x,y) = \sin(\pi x)\sin(\pi y), \qquad f(x,y) = 2\pi^2 \sin(\pi x)\sin(\pi y).$$

The discretization is carried out using the **newly constructed quadrilateral finite element**, based on the isoparametric transformation of Figure 5. We consider uniform meshes of size $n \times n$ for $n = 2, 4, 8, 16$.

The numerical errors in L^2 and H^1 norms are reported in Table 1.

n	$\|u-u_h\|_{L^2}$	Order L^2	$\|u-u_h\|_{H^1}$	Order H^1
2	1.23e-02	–	3.41e-01	–
4	1.55e-03	2.99	8.57e-02	1.99
8	1.95e-04	2.99	2.14e-02	2.00
16	2.44e-05	3.00	5.36e-03	2.00

TABLE 1. L^2 and H^1 errors for the Poisson problem with our quadrilateral finite element (Figure 5).

These results confirm that the proposed quadrilateral element indeed achieves the theoretical convergence orders: $\mathcal{O}(h^3)$ in the L^2 norm and $\mathcal{O}(h^2)$ in the H^1 norm. This validates the consistency and efficiency of our new construction.

4.2. **Convection–Diffusion Equation Problem.** We consider the time-dependent convection–diffusion equation

$$\frac{\partial u}{\partial t} + v \cdot \nabla u = K \Delta u + f \quad \text{in} \quad \Omega \times (0,T),$$

subject to the initial condition

$$u(x,y,0) = u_0(x,y), \quad (x,y) \in \Omega,$$

and the homogeneous Dirichlet boundary condition

$$u = 0 \quad \text{on } \partial\Omega \times (0,T).$$

The exact analytical solution chosen for validation is

$$u^{\text{ex}}(x,y,t) = a\, e^{bt} \left(e^{-cx} + e^{-cy}\right),$$

defined in the square domain $\Omega = [-1,1]^2$, with the parameters

$$K = 1, \quad v = (1,1), \quad a = 1, \quad b = 0.1, \quad c = \tfrac{1+\sqrt{1.4}}{2}.$$

Temporal discretization is performed using an implicit backward Euler scheme, ensuring unconditional stability, while spatial discretization is based on the newly constructed quadrilateral finite element (see Figure 5), employing the same isoparametric transformation as for the Poisson problem. This formulation allows an accurate representation of both the diffusion and convection effects.

Description of the RBF method used for comparison. For benchmarking purposes, a mesh-free **Radial Basis Function (RBF)** approach was also implemented to solve the same problem under identical physical and numerical settings. The approximate solution is expressed as

$$u(x,y,t) = \sum_{i=1}^{N} \lambda_i(t)\, \phi(\|(x,y) - (x_i,y_i)\|)\,,$$

where $\{(x_i,y_i)\}_{i=1}^N$ are collocation nodes uniformly distributed in Ω, $\lambda_i(t)$ are the unknown time-dependent coefficients, and $\phi(r)$ denotes the selected radial basis function.

In this work, the **Multiquadric (MQ)** function

$$\phi(r) = \sqrt{r^2 + c^2},$$

was adopted, with the shape parameter $c = 0.3\,h$, where h denotes the average distance between neighboring nodes.

The governing equation and boundary conditions were enforced at all collocation points following the method of lines. Spatial derivatives ∇u and Δu were computed analytically from the RBF expressions, and the resulting semi-discrete system of ordinary differential equations was integrated in time using the same implicit scheme as for the finite element method to ensure a fair and consistent comparison.

Homogeneous Dirichlet boundary conditions were imposed by excluding the boundary nodes from the collocation set. The same analytical solution $u^{\text{ex}}(x, y, t)$ and parameters were used to evaluate the numerical error at each time level.

Error comparison and analysis. The maximum absolute error is defined as

$$E = \max_{(x,y)\in\Omega} |u(x, y, t) - u^{\text{ex}}(x, y, t)|.$$

Table 2 presents the results obtained using our quadrilateral finite element (E_Q) and the RBF-based method (E_{RBF}) for different mesh sizes h and times t.

h	t	E_Q	E_{RBF}
0.11	$t = 0$	1.450×10^{-3}	1.100×10^{-2}
	$t = 0.2$	2.013×10^{-3}	1.600×10^{-2}
	$t = 0.4$	2.874×10^{-3}	1.900×10^{-2}
0.05	$t = 0$	2.021×10^{-4}	1.600×10^{-3}
	$t = 0.2$	4.001×10^{-4}	2.600×10^{-3}
	$t = 0.4$	4.201×10^{-4}	3.200×10^{-3}
0.03	$t = 0$	5.111×10^{-5}	1.345×10^{-4}
	$t = 0.2$	8.120×10^{-5}	1.901×10^{-4}
	$t = 0.4$	6.338×10^{-5}	2.322×10^{-4}

TABLE 2. Maximum absolute error between the numerical solution $u(x, y, t)$ and the analytical solution $u^{\text{ex}}(x, y, t)$ for different mesh sizes h and times t, obtained with the proposed quadrilateral finite element and the RBF method.

The results show that the proposed quadrilateral finite element consistently achieves higher accuracy than the RBF-based method across all test cases. The error decreases rapidly with mesh refinement, demonstrating the robustness, precision, and numerical efficiency of the proposed formulation for solving convection–diffusion problems. Moreover, the lower error levels observed even for coarse meshes highlight the superior approximation capability of the new quadrilateral element compared to mesh-free RBF approaches.

5. Conclusion and Future Work

To solve two-dimensional convection-diffusion partial differential equations (PDEs), this work introduces a novel quadrilateral finite element constructed from quadratic cardinal B-splines. The development starts by building the element through tensor-product techniques applied in both one- and two-dimensional contexts. Numerical tests demonstrate that the method attains a satisfactory convergence rate, indicating its effectiveness for approximating such PDEs.

The method was evaluated on a convection-diffusion equation with constant coefficients, subject to Dirichlet boundary conditions on a quadrangulated domain. The maximum error between the numerical and exact solutions demonstrated improved accuracy compared to methods based on radial basis functions (RBFs).

The construction presented here is straightforward and opens several promising directions for future work. Potential extensions include the development of trivariate splines on hexahedral-tetrahedral meshes, the incorporation of local refinement to enhance flexibility, and the generalization of the method to higher polynomial degrees.

References

1. Abbadi, A., Ibànez, M. J., & Sbibih, D. (2011). Computing quasi-interpolants from the B-form of B-splines. *Mathematics and Computers in Simulation, 81*, 1936–1948. https://doi.org/10.1016/j.matcom.2011.03.020
2. Bazilevs, Y., Beirão da Veiga, L., Cottrell, J. A., Hughes, T. J. R., & Sangalli, G. (2006). Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences, 16*(7), 1031–1090. https://doi.org/10.1142/S021820250600093X
3. Babuska, I., & Rheinboldt, W. C. (1991). Review of some adaptive node-movement techniques in finite-element and finite-difference solutions of partial differential equations. *Journal of Computational Physics, 95*(2), 254–302. https://doi.org/10.1016/0021-9991(91)90123-T
4. Braess, D. (2007). *Finite elements: Theory, fast solvers, and applications in solid mechanics* (3rd ed.). Cambridge University Press.
5. Barrera, D., Guessab, A., Ibànez, M. J., & Nouisser, O. (2010). Optimal bivariate C^1 cubic quasi-interpolation on a type-2 triangulation. *Journal of Computational and Applied Mathematics, 234*, 1188–1199. https://doi.org/10.1016/j.cam.2009.12.009
6. Cottrell, J. A., Hughes, T. J. R., & Bazilevs, Y. (2009). *Isogeometric analysis: Toward integration of CAD and FEA*. Wiley.
7. de Boor, C. (2001). *A Practical Guide to Splines* (Revised edition, Chapters II–III). Springer. https://doi.org/10.1007/978-1-4612-2262-1
8. Douglas, J., Todd, D., Dupont, P., & Ridgway-Scott, S. (1979). A family of C^1 finite elements with optimal approximation properties for various Galerkin methods for 2nd and 4th order problems. *RAIRO: Analyse Numérique, 13*(3), 227–255.
9. Hughes, T. J. R. (2000). *The finite element method: Linear static and dynamic finite element analysis*. Dover.
10. Lamnii, A., Lamnii, M., & Mraoui, H. (2015). Constructing B-spline representation of quadratic Sibson–Thomson splines. *Computer Aided Geometric Design, 33*, 66–81. https://doi.org/10.1016/j.cagd.2015.01.004
11. Lamnii, A., Lamnii, M., & Oumellal, F. (2017). Computation of Hermite interpolation in terms of B-spline basis using polar forms. *Mathematics and Computers in Simulation, 134*, 17–27. https://doi.org/10.1016/j.matcom.2017.04.001

12. Lamnii, M., Mraoui, H., Tijini, A., & Zidna, A. (2014). A normalized basis for C^1 cubic super spline space on Powell-Sabin triangulation. *Mathematics and Computers in Simulation, 99*, 108–124. https://doi.org/10.1016/j.matcom.2013.10.017
13. Lamnii, M., Mraoui, H., & Tijini, A. (2013). Construction of quintic Powell-Sabin spline quasi-interpolants based on blossoming. *Journal of Computational and Applied Mathematics, 250*, 190–209. https://doi.org/10.1016/j.cam.2012.09.031
14. Lamnii, M., Mraoui, H., & Tijini, A. (2013). Local quasi-interpolants based on special multivariate quadratic spline space over a refined quadrangulation. *Applied Mathematics and Computation, 219*, 10456–10467. https://doi.org/10.1016/j.amc.2013.06.053
15. Dagnino, C., Lamberti, P., & Remogna, S. (2012). Trivariate quartic splines on type-6 tetrahedral partitions and their applications. *Communications in Applied and Industrial Mathematics, 3*, 1–9.
16. Oudin, H. (2008). *Méthode des éléments finis* (pp. 63). École Centrale de Nantes. https://hal.archives-ouvertes.fr/hal-00341772
17. Rahouti, A., Serghini, A., & Tijini, A. (n.d.). Construction of a normalized basis of a univariate quadratic C^1 spline space and application to the quasi-interpolation. *Manuscript submitted for publication.*
18. Speleers, H. (2012). Construction of normalized B-splines for a family of smooth spaces over Powell-Sabin triangulations. *Constructive Approximation, 37*, 41–72. https://doi.org/10.1007/s00365-012-9163-0
19. Speleers, H. (2013). Multivariate normalized Powell-Sabin B-splines and quasi-interpolants. *Computer Aided Geometric Design, 30*, 2–19. https://doi.org/10.1016/j.cagd.2012.09.003
20. Zienkiewicz, O. C., Taylor, R. L., & Zhu, J. Z. (2005). *The finite element method: Its basis and fundamentals* (6th ed.). Elsevier.

[1] Hassan First University, Laboratory of Education, Science and Technology, ESEF Berrechid, Morocco.
Email address: fatima.oumellal@uhp.ac.ma; https://orcid.org/0009-0006-4429-4196

[2] University Mohamed First, Laboratory LANO, Faculty of Science Oujda, Oujda, Morocco.
Email address: sheimamouhoub@gmail.com; https://orcid.org/0009-0007-6209-5307

[2] University Mohamed First, Laboratory LANO, Faculty of Science Oujda, Oujda, Morocco.
Email address: m_lamnii1@yahoo.fr; https://orcid.org/0000-0002-2532-3418.,

[3] Abdelmalek Essaâdi University, Laboratory ISI, ENSATe, 93030, Tétouan, Morocco.
Email address: a.lamnii@uae.ac.ma; https://orcid.org/0000-0002-0538-8812.,