J. Math. Computer Sci., 36 (2025), 408–431

Online: ISSN 2008-949X

Journal of Mathematics and Computer Science

Journal Homepage: www.isr-publications.com/jmcs

An advanced numerical technique for subdivision depth of non-stationary quaternary refinement scheme for curves and surfaces

Ayesha Mahmood, Ghulam Mustafa, Faheem Khan

Department of Mathematics, The Islamia University of Bahawalpur, Pakistan.

Abstract

Refinement schemes are fundamental in computer graphics for generating smooth curves and surfaces. Quaternary non-stationary subdivision schemes, in particular, have gained prominence due to their ability to handle complex geometric structures. However, determining the subdivision depth for these schemes remains challenging and often requires extensive computational resources. Our paper presents a complete methodology with a step-by-step explanation to explore the depth of these schemes. Since our method relies on convolution techniques, we explain these both theoretically and mathematically. Additionally, several algorithms have been designed to aid in understanding and implementing the method for finding error bounds and subdivision depth in quaternary non-stationary subdivision schemes. These are numerical methods for efficiently computing the error bounds and subdivision depth. The numerical applications of these methods are presented. The proposed method significantly reduces the computational cost associated with determining subdivision depth. These algorithms work when existing methods fail to compute bounds and depths.

Keywords: Convolution, error bound, subdivision depth, quaternary non-stationary refinement schemes, algorithm.

2020 MSC: 65D17, 65D05, 65D10, 65Y04.

1. Introduction

Subdivision schemes are a class of algorithms used for generating smooth curves and surfaces by iteratively refining an initial control mesh. The basic idea behind subdivision schemes [10] is to work on the principle that a simple initial mesh is used as a starting point and then refined using a set of rules or algorithms. These schemes are essential tools in the field of computer graphics and computational geometry, providing a way to represent smooth curves and surfaces with a relatively low initial resolution and gradually refining them for more realistic and detailed results [11]. These schemes are classified as stationary and non-stationary [2–6, 9], uniform and non-uniform [13, 14], and linear and non-linear [12] schemes of any arity. The arity of the subdivision scheme is the number of points inserted at level $\ell+1$ between two consecutive level ℓ points. The refinement schemes are known as binary, ternary, and

Email addresses: ayesha.mehmood@iub.edu.pk (Ayesha Mahmood), ghulam.mustafa@iub.edu.pk (Ghulam Mustafa), faheem.khan@iub.edu.pk (Faheem Khan)

doi: 10.22436/jmcs.036.04.05

Received: 2024-03-24 Revised: 2024-06-01 Accepted: 2024-07-17

quaternary, respectively, when there are 2, 3, and 4 points inserted. The term "quaternary" refers to the fact that the subdivision scheme inserts four new points at each refinement level between every old consecutive pair of points of the previous refinement level. Quaternary subdivision schemes refer to the subdivision process where each control point is replaced by four new points in each iteration. This means that the number of control points quadruples at every iteration, leading to a finer mesh and smooth curves or surfaces. For more detail on stationary and non-stationary quaternary subdivision schemes, we may refer to [1, 25, 28]. In non-stationary quaternary schemes, the refinement rules can vary based on factors like curvature, tangent direction, or other local geometric properties. This flexibility allows for more control over the shape of the resulting curves and surfaces.

Error bounds (the difference between the subdivision-generated limit surface and the surface at the ℓ^{th} level) and subdivision depth (the number of iterations to get the desired shape) are fundamental concepts in Computer-Aided Geometric Design (CAGD), a field that deals with the mathematical and computational aspects of creating, manipulating, and representing geometric shapes. These concepts are especially important in the context of curve and surface approximation and refinement. When working with curves or surfaces, CAGD practitioners often need to approximate complex shapes with simpler, more manageable representations. Error bounds allow them to control the trade-off between computational efficiency and accuracy. Error bounds and subdivision depth are important concepts in the context of non-stationary quaternary subdivision schemes. These schemes are used in computer graphics and image processing to refine curves or surfaces by iteratively subdividing them into smaller ones.

Subdivision depth in non-stationary quaternary subdivision schemes has various applications in engineering, particularly in fields such as Computer-Aided Design (CAD), image processing, and geometry modeling. These schemes are used to refine curves and surfaces in a progressive and controlled manner. Subdivision depth is vital in the generation and refinement of 3D meshes, which are used in finite element analysis [7, 18, 30], computational fluid dynamics, and other simulations. By controlling the subdivision depth, engineers can balance computational efficiency with the accuracy of their simulations.

In product design, achieving smooth and accurate curves and surfaces is essential. This method can help product designers determine the optimal level for computer-aided geometric models. Engineers working on mechanical components [26] can benefit from precise control over subdivision depths to ensure that critical features are accurately represented. Engineers designing aircraft and spacecraft [29] can use this method for creating aerodynamic shapes and surfaces. Engineers involved in civil engineering projects, such as bridges and tunnels, can use this method to control the subdivision depth and achieve realistic visualizations. Engineers designing medical devices and implants can benefit from precise modeling to ensure a proper fit and function. This method can help achieve the required accuracy. In environmental engineering applications, accurate terrain modeling is crucial. This method can be used to optimize the representation of terrain data for analysis [19]. Several authors have calculated error bounds for various subdivision models, including n-ary [22], tensor product n-ary volumetric [23], and binary volumetric [24] models. These computations typically rely on the first forward difference and often result in broad error bounds. However, these methods are not universally applicable to all types of subdivision schemes, particularly higher arity non-stationary schemes. The proposed method addresses this gap by proposing an optimal method to estimate subdivision depths specifically for quaternary non-stationary subdivision schemes.

In recent years, significant research has been dedicated to estimating error bounds [8, 15, 20, 21, 31] and determining subdivision depth [17, 27] for both stationary and non-stationary subdivision schemes. Surprisingly, there has been a notable gap in the existing literature regarding the computation of subdivision depth and error bounds for high-arity non-stationary subdivision schemes. In the current work, we focus on computing error bounds and subdivision depth for quaternary non-stationary refinement schemes, both univariate and bivariate, in relation to their control polygons. We introduce a novel method based on the convolution of two vectors to precisely determine these error bounds and subdivision depths for quaternary non-stationary refinement schemes.

1.1. *Methodology*

Here, we design a comprehensive methodology for estimating the error bounds and subdivision depth of non-stationary quaternary refinement schemes for curves and surfaces, which involves several steps.

- Problem definition: In quaternary subdivision schemes each edge of initial control polygon is divided into 4 sub-edges. Non-stationary schemes are level-dependent and are more appropriate for generating conic sections. The estimation of error bounds and subdivision depth for specific quaternary non-stationary subdivision schemes, which are based on convolution, is the main problem of our work.

- Initial data: The initial data serves as the starting point for the subdivision process. It consists of control points and their weights. The subdivision process refines these control points and weights to achieve the desired level of smoothness and detail. As the subdivision level increases, the initial data are iteratively refined, and more detailed approximations of the original shape are generated.

- Convolution: Consider the vector $J_i = J_i^\ell$, at the ℓ^{th} level of resolution, represents the approximation coefficients. Then the reconstruction algorithm of non-stationary quaternary refinement scheme used to describe the approximation coefficient of two consecutive stages ℓ and $\ell+1$ is defined as

$$J_i^{\ell+1} = \sum_{n\in\mathbb{N}} J_n^\ell \hat{h}_{i-4n}^\ell = (J_n^{\ell;0} \star \hat{h}^\ell)_j,$$

where $J_i^0 = J_i^{\ell;0}$ represents the ℓth resolution and $\star$ denotes the convolution of two vectors $J_n^{\ell;0}$ and $\hat{h}^\ell = (\hat{h}_n^\ell)_{n\in\mathbb{N}}$. Generally, for finite lengths l_J and $l_{\hat{h}}$ the convolution of two vectors $J^\ell = (J_n^\ell)_{n\geqslant 0}$ and $\hat{h}^\ell = (\hat{h}_n^\ell)_{n\geqslant 0}$ can be defined as

$$(J^{\ell;0} \star \hat{h}^\ell)_j = \sum_{n=\max\{j-(l_{\hat{h}}-1),0\}}^{\min\{j,l_J-1\}} J_n^{\ell;0}\hat{h}_{j-4n}^\ell, j = 0, 1, \dots, l_J + l_{\hat{h}} - 2.$$

- Convolution coefficient: Here, first we assign a sequence of constants as vectors $q_{0,\hat{s}}^\ell$, $q_{1,\hat{s}}^\ell$, $q_{2,\hat{s}}^\ell$, and $q_{3,\hat{s}}^\ell$ using methodology given in [27]. With the help of these real valued constants, we estimate $T_{\gamma_0^c}$ for the univariate case and $M_{\gamma_0^s}N_{\gamma_0^s}$ for the bivariate case. Here, the notations γ_0^c and γ_0^s are used for the order of convolution, where c and s are used for the curve and surface cases, respectively. These constants are monotonically decreasing corresponding to the increase of subdivision level using convolution. This is the main goal of our proposed work.

- Error estimation framework: We establish a theoretical framework for computing error bounds for curves and surfaces during the subdivision process. We utilize mathematical analysis to derive expressions that quantify the difference between the original curve/surface and the subdivided curve/surface. Specifically, we determine the error bounds between two consecutive control polygons of quaternary non-stationary subdivision schemes, i.e., $||\zeta^{\ell+1} - \zeta^\ell||_\infty$, and then between limit subdivision curves/surfaces and their control polygons, i.e., $||\zeta^\infty - \zeta^\ell||_\infty$. Here, ζ^ℓ is the curve/surface obtained after ℓ^{th} iteration, and ζ^∞ is the limit curve/surface.

- Convolution-based subdivision depth determination: We employ the methodology outlined in [27] to ascertain the optimal subdivision depth for curves and surfaces, taking into account factors such as target error tolerance and geometric complexity. Additionally, we provide algorithms or guidelines for estimating the subdivision depth based on user-defined error tolerance.

- Selection of subdivision scheme: We select interpolating [28] and approximating [1] quaternary non-stationary subdivision schemes as the focus of the research. The selection is based on their relevance, uniqueness, and potential benefits.

Furthermore, in the convolution process, we use a vector to represent approximation coefficients at a given resolution level. The non-stationary quaternary refinement scheme describes these coefficients between two consecutive stages by summing the products of the current level's coefficients with weight functions. Convolution involves summing the products of two vectors' elements with appropriately shifted elements of the other vector. To calculate convolution coefficients, we assign a sequence of constants as vectors, following a specific methodology. These constants help estimate values for univariate and bivariate cases, distinguishing between curves and surfaces. They decrease monotonically with increasing subdivision levels. Some algorithms for implementing these methods are designed and presented in Section 5.

The paper is organized as follows. In Section 2, we discuss definitions of non-stationary quaternary subdivision schemes, their error bounds, and subdivision depths for both univariate and bivariate cases. Section 3 presents the main results, numerical applications, and graphical representations for the univariate case. Section 4 does the same for the bivariate case. Section 5 outlines algorithms for convolution, error bounds, and subdivision depth. Finally, Section 6 presents the conclusion.

2. Preliminaries

Here, we demonstrate notations, main results of convolution, error bounds, subdivision depth, and numerical experiments for both curve and surface models.

2.1. *Quaternary non-stationary univariate case*

In a subdivision scheme, a set of control points at level ℓ is given by $\{J_i^\ell \in \mathbb{R}^{\mathbb{N}} \mid i \in \mathbb{Z}, \mathbb{N} \geqslant 2\}$, where $\ell \geqslant 0$ denotes the subdivision level or iteration level. The refinement process generates new points at the $(\ell+1)$th level, which are used in subsequent iterations to further refine the curve/surface. The general form of a non-stationary quaternary subdivision scheme (NSQSS) can be expressed as:

$$J_{4i+\hat{n}}^{\ell+1} = \sum_{\hat{s}=0}^{P-1} c_{\hat{n},\hat{s}}^{\ell} J_{i+\hat{s}}^{\ell}, \quad \hat{n} = 0,1,2,3, \tag{2.1}$$

with a necessary condition for the uniform convergence, where P is an integer greater than zero,

$$\sum_{\hat{s}=0}^{P-1} c_{\hat{n},\hat{s}}^{\ell} = 1, \quad \hat{n} = 0,1,2,3, \tag{2.2}$$

where $\{c_{\hat{n},\hat{s}}^{\ell},\ \hat{n} = 0,1,2,3\}_{\hat{s}=0}^{P-1}$ is the set of coefficients are usually called the mask of subdivision scheme at the ℓ^{th} level. The scheme is classified as non-stationary if the mask depends on ℓ; otherwise, it is a stationary scheme. The set of initial control polygons can be expressed as $\{J_i^0 \in \mathbf{R}^{\mathbb{N}}, i \in \mathbb{Z}\}$, where $\mathbb{N} \geqslant 2$. Then in the limit $\ell \to \infty$, the procedure describes an infinite set of points in $\mathbf{R}^{\mathbb{N}}$. The sequence of control points $\{J_i^\ell\}$ is associated with the diadic mesh point $f_i^\ell = \frac{i}{4^\ell}, i \in \mathbb{Z}$. The formulation (2.1) specify a scheme by which $J_{4i}^{\ell+1}$ replaces/takeover the value J_i^ℓ at the mesh point $f_{4i}^{\ell+1} = f_i^\ell$, while $J_{4i+1}^{\ell+1}$,$J_{4i+2}^{\ell+1}$, and $J_{4i+3}^{\ell+1}$ are inserted at the new mesh points $J_{4i+1}^{\ell+1} = \frac{1}{4}(3f_i^\ell + f_{i+1}^\ell)$, $J_{4i+2}^{\ell+1} = \frac{1}{2}(f_i^\ell + f_{i+1}^\ell)$, and $J_{4i+3}^{\ell+1} = \frac{1}{4}(f_i^\ell + 3f_{i+1}^\ell)$, respectively.

2.2. *Error bounds and subdivision depth for non-stationary quaternary subdivision schemes*

In this subsection, we establish error bounds between two consecutive control polygons of NSQSS, namely $\|\chi^{\ell+1} - \chi^\ell\|_\infty$ and error bounds between limit subdivision curves/surfaces and their control polygons, i.e., $\|\chi^\infty - \chi^\ell\|_\infty$, where χ^ℓ is the control polygon obtained after ℓ^{th} level and χ^∞ is the limiting curve. We utilize the results from [16], which generalize those in [20]. These generalized results are then applied to the 4-point approximating scheme [1] and the 4-point interpolating scheme [28] of NSQSS to derive error bounds. Now, for estimating error bounds and determining the subdivision depth of NSQSS, we adjust the coefficients as follows:

$$\begin{aligned} q^{\ell}_{0,\hat{s}} &= \sum_{y=0}^{\hat{s}} \left(c^{\ell}_{0,y} - c^{\ell}_{1,y}\right), \quad q^{\ell}_{1,\hat{s}} = \sum_{y=0}^{\hat{s}} \left(c^{\ell}_{1,y} - c^{\ell}_{2,y}\right), \\ q^{\ell}_{2,\hat{s}} &= \sum_{y=0}^{\hat{s}} \left(c^{\ell}_{2,y} - c^{\ell}_{3,y}\right), \quad q^{\ell}_{3,\hat{s}} = c^{\ell}_{0,\hat{s}} - (q^{\ell}_{0,\hat{s}} + q^{\ell}_{1,\hat{s}} + q^{\ell}_{2,\hat{s}}), \end{aligned} \tag{2.3}$$

with the strong condition provided in [16],

$$\sum_{\hat{s}=0}^{P-1} |q^{\ell}_{0,\hat{s}}| < 1, \quad \sum_{\hat{s}=0}^{P-1} |q^{\ell}_{1,\hat{s}}| < 1, \quad \sum_{\hat{s}=0}^{P-1} |q^{\ell}_{2,\hat{s}}| < 1, \quad \sum_{\hat{s}=0}^{P-1} |q^{\ell}_{3,\hat{s}}| < 1.$$

In compact form, equation (2.3) can be written as

$$\begin{cases} q_{\hat{b},\hat{s}} = \sum\limits_{y=0}^{\hat{s}} (c^{\ell}_{\hat{b},y} - c^{\ell}_{\hat{b}+1,y}), \ \hat{b} = 0,1,2, \\ q_{3,\hat{s}} = c^{\ell}_{0,\hat{s}} - \sum\limits_{\hat{b}=0}^{2} q_{\hat{b},\hat{s}}. \end{cases}$$

Now, let's assign new symbols for $\hat{s} = 0, 1, \dots, P-1$, such that

$$\hat{h}^{\ell}_{4\hat{s}} = q^{\ell}_{0,\hat{s}}, \quad \hat{h}^{\ell}_{4\hat{s}+1} = q^{\ell}_{1,\hat{s}}, \quad \hat{h}^{\ell}_{4\hat{s}+2} = q^{\ell}_{2,\hat{s}}, \quad \hat{h}^{\ell}_{4\hat{s}+3} = q^{\ell}_{3,\hat{s}}. \tag{2.4}$$

In the next step, we will continue and provide some convolutional results for a one-dimensional array of vectors dependent on NSQSS.

Lemma 2.1. *Let $\{J^{\ell}_n; n \geqslant 0\}$ represents the vector of finite length and $\{\hat{h}^{\ell}_n; n \geqslant 0\} = (\hat{h}^{\ell}_n)_{n=0}^{4P-1}$ with $\hat{h}^{\ell}_n = 0$ for $n \geqslant 4P$. Then one dimensional convolution between $J = J^{\ell}_n$ and $\hat{h} = \hat{h}^{\ell}_n$ for NSQSS for curve can be expressed as*

$$((J^{(0)})^{\ell} \star \hat{h}^{\ell})_j = \sum_{n=0}^{\lfloor j/4 \rfloor} J^{\ell}_n \hat{h}^{\ell}_{j-4n}. \tag{2.5}$$

Similarly, the γ^{c}_0-times convolution reformulation is described as

$$((\cdots((((J^{(0)})^{\ell} \star \hat{h}^{\ell})^{(0)}) \star \hat{h}^{\ell})^{(0)} \star \cdots \star \hat{h}^{\ell})^{(0)} \star \hat{h}^{\ell})_j = \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma^{c}_0} \rfloor} J^{\ell}_{\hat{s}} S^{\gamma^{c}_0;\hat{h}^{\ell}}_{\hat{s},j}, \tag{2.6}$$

with

$$S^{1;\hat{h}^{\ell}}_{\hat{s},j} = \hat{h}^{\ell}_{j-4\hat{s}} \quad \textit{and} \quad S^{\gamma^{c}_0;\hat{h}^{\ell}}_{\hat{s},j} = \sum_{J=4\hat{s}}^{\lfloor j/4^{\gamma^{c}_0-1} \rfloor} S^{1;\hat{h}^{\ell}}_{\hat{s},J} S^{\gamma^{c}_0-1;\hat{h}^{\ell}}_{J,j}, \qquad \gamma^{c}_0 \geqslant 2.$$

Hence by (2.6), we get

$$\|((\cdots((((J^{(0)})^{\ell} \star \hat{h}^{\ell})^{(0)}) \star \hat{h}^{\ell})^{(0)} \star \cdots \star \hat{h}^{\ell})^{(0)} \star \hat{h}^{\ell})\|_{\infty} \leqslant \|J^{\ell}\|_{\infty} \max_j \left\{ \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma^{c}_0} \rfloor} |S^{\gamma^{c}_0;\hat{h}^{\ell}}_{\hat{s},j}| \right\}, \tag{2.7}$$

where the associated constant of the γ_0^c*-times convolution for NSQSS can be specify as:*

$$T_{\gamma_0^c} = \max_j \left\{ \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c} \rfloor} |S_{\hat{s},j}^{\gamma_0^c;\hat{h}^\ell}| \right\} = \max_{j\in\Sigma(\gamma_0^c,P)} \left\{ \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c} \rfloor} |S_{\hat{s},j}^{\gamma_0^c;\hat{h}^\ell}| \right\}, \tag{2.8}$$

with

$$\Sigma(\gamma_0^c, P) = \{\Omega(\gamma_0^c, P) - 4^{\gamma_0^c} + 1, \Omega(\gamma_0^c, P) - 4^{\gamma_0^c} + 2, \dots, \Omega(\gamma_0^c, P)\}, \tag{2.9}$$

and

$$\Omega(\gamma_0^c, P) = (4^{\gamma_0^c} - 3)(4P - 1). \tag{2.10}$$

Proof. Now, we begin with the cases of $\gamma_0^c = 1$ and $\gamma_0^c = 2$ convolutions, and then we will develop the general case.

Case $\gamma_0^c = 1$: From (2.5), we acquire a relation given in the following

$$((J^{(0)})^\ell \star \hat{h}^\ell)_j = \sum_{n=0}^{\lfloor j/4 \rfloor} J_n^\ell \hat{h}_{j-4n}^\ell, \tag{2.11}$$

where $\lfloor . \rfloor$ represent the integer part. By taking norm, we have

$$|((J^{(0)})^\ell \star \hat{h}^\ell)_j| = \left| \sum_{n=0}^{\lfloor j/4 \rfloor} J_n^\ell \hat{h}_{j-4n}^\ell \right| \leqslant \sum_{n=0}^{\lfloor j/4 \rfloor} |J_n^\ell| \sum_{n=0}^{\lfloor j/4 \rfloor} |\hat{h}_{j-4n}^\ell|.$$

By using infinity norm ($\|J^\ell\|_\infty = \max\{|J_0^\ell|, \dots, |J_{\lfloor j/4 \rfloor}^\ell|\}$), we have

$$|((J^{(0)})^\ell \star \hat{h}^\ell)_j| \leqslant \|J^\ell\|_\infty \sum_{n=0}^{\lfloor j/4 \rfloor} |\hat{h}_{j-4n}^\ell|.$$

Now

$$\max |((J^{(0)})^\ell \star \hat{h}^\ell)_j| \leqslant \max \left(\|J^\ell\|_\infty \sum_{n=0}^{\lfloor j/4 \rfloor} |\hat{h}_{j-4n}^\ell| \right).$$

We drive

$$\max |((J^{(0)})^\ell \star \hat{h}^\ell)_j| \leqslant \|J^\ell\|_\infty \max \left(\sum_{n=0}^{\lfloor j/4 \rfloor} |S_{n,j}^{1;\hat{h}^\ell}| \right),$$

where

$$\hat{h}_{j-4n}^\ell = S_{n,j}^{1;\hat{h}^\ell}.$$

This implies

$$|((J^{(0)})^\ell \star \hat{h}^\ell)_j|_\infty \leqslant \|J^\ell\|_\infty \max \left(\sum_{n=0}^{\lfloor j/4 \rfloor} |S_{n,j}^{1;\hat{h}^\ell}| \right).$$

Case $\gamma_0^c = 2$: From (2.11), we obtain

$$(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = \sum_{\hat{s}=0}^{\lfloor j/4 \rfloor} ((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)_{\hat{s}} \hat{h}^\ell_{j-4\hat{s}} = \sum_{\hat{s}=0}^{\lfloor j/4 \rfloor} \left(\sum_{n=0}^{\lfloor \hat{s}/4 \rfloor} \mathrm{J}^\ell_n \hat{h}^\ell_{\hat{s}-4n} \right) \hat{h}^\ell_{j-4\hat{s}},$$

which drives

$$\begin{aligned}(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = {} & \mathrm{J}^\ell_0 (\hat{h}^\ell_0 \hat{h}^\ell_j + \hat{h}^\ell_1 \hat{h}^\ell_{j-4} + \hat{h}^\ell_2 \hat{h}^\ell_{j-8} + \hat{h}^\ell_3 \hat{h}^\ell_{j-12} + \hat{h}^\ell_4 \hat{h}^\ell_{j-16} + \cdots + \hat{h}^\ell_{\lfloor \frac{j}{4} \rfloor} \hat{h}^\ell_0) \\ & + \mathrm{J}^\ell_1 (\hat{h}^\ell_0 \hat{h}^\ell_{j-16} + \hat{h}^\ell_1 \hat{h}^\ell_{j-20} + \cdots + \hat{h}^\ell_0 \hat{h}^\ell_{\lfloor \frac{j}{4} \rfloor - 4}) + \mathrm{J}^\ell_2 (\hat{h}^\ell_0 \hat{h}^\ell_{j-32} + \hat{h}^\ell_1 \hat{h}^\ell_{j-36} \\ & + \cdots + \hat{h}^\ell_0 \hat{h}^\ell_{\lfloor \frac{j}{4} \rfloor - 8}) + \cdots + \mathrm{J}^\ell_{\lfloor \frac{j}{4^2} \rfloor} \hat{h}^\ell_0 \hat{h}^\ell_0.\end{aligned}$$

This implies

$$\begin{aligned}(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = {} & \mathrm{J}^\ell_0 \left(\sum_{n=0}^{\lfloor j/4 \rfloor} \hat{h}^\ell_n \hat{h}^\ell_{j-4n} \right) + \mathrm{J}^\ell_1 \left(\sum_{n=4}^{\lfloor j/4 \rfloor} \hat{h}^\ell_{n-4} \hat{h}^\ell_{j-4n} \right) + \mathrm{J}^\ell_2 \left(\sum_{n=8}^{\lfloor j/4 \rfloor} \hat{h}^\ell_{n-8} \hat{h}^\ell_{j-4n} \right) \\ & + \cdots + \mathrm{J}^\ell_{\lfloor j/4^2 \rfloor} \left(\sum_{n=4\lfloor \frac{j}{4^2} \rfloor}^{\lfloor j/4 \rfloor} \hat{h}^\ell_{n-4\lfloor \frac{j}{4^2} \rfloor} \hat{h}^\ell_{j-4n} \right).\end{aligned}$$

This further implies

$$(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = \sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} (\mathrm{J}_{\hat{s}})^\ell \left(\sum_{n=4\hat{s}}^{\lfloor j/4 \rfloor} \hat{h}^\ell_{n-4\hat{s}} \hat{h}^\ell_{j-4n} \right).$$

We acquire

$$(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = \sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} (\mathrm{J}_{\hat{s}})^\ell \left(\sum_{n=4\hat{s}}^{\lfloor j/4 \rfloor} S^{1;\hat{h}^\ell}_{\hat{s},n} S^{1;\hat{h}^\ell}_{n,j} \right) = \sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} (\mathrm{J}_{\hat{s}})^\ell S^{2;\hat{h}^\ell}_{\hat{s},j},$$

where

$$S^{2;\hat{h}^\ell}_{\hat{s},j} = \sum_{n=4\hat{s}}^{\lfloor j/4 \rfloor} S^{1;\hat{h}^\ell}_{\hat{s},n} S^{1;\hat{h}^\ell}_{n,j}. \tag{2.12}$$

We get

$$|(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j| = \left| \sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} (\mathrm{J}_{\hat{s}})^\ell S^{2;\hat{h}^\ell}_{\hat{s},j} \right| \leqslant \|\mathrm{J}^\ell\|_\infty \sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} |S^{2;\hat{h}^\ell}_{\hat{s},j}|.$$

This implies

$$\|(((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j\|_\infty \leqslant \|\mathrm{J}^\ell\|_\infty \max \left(\sum_{\hat{s}=0}^{\lfloor j/4^2 \rfloor} |S^{2;\hat{h}^\ell}_{\hat{s},j}| \right).$$

General case: The reformulations for the γ_0^c-th convolutions are obtained by applying the same method, which is as follows:

$$((\cdots((((\mathrm{J}^{(0)})^\ell \star \hat{h}^\ell)^{(0)}) \star \hat{h}^\ell)^{(0)} \star \cdots \star \hat{h}^\ell)^{(0)} \star \hat{h}^\ell)_j = \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c} \rfloor} \mathrm{J}^\ell_{\hat{s}} S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s},j}.$$

Hence, we get

$$\|((\cdots((((J^{(0)})^{\ell}\star\hat{h}^{\ell})^{(0)})\star\hat{h}^{\ell})^{(0)}\star\cdots\star\hat{h}^{\ell})^{(0)}\star\hat{h}^{\ell})\|_{\infty}\leqslant\|J^{\ell}\|_{\infty}\max_{j}\left\{\sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c}\rfloor}|S_{\hat{s},j}^{\gamma_0^c;\hat{h}^{\ell}}|\right\}.$$

□

Lemma 2.2. *The term* $S_{\hat{s},j}^{\gamma_0^c;\hat{h}^{\ell}}$ *in* (2.7) *has the following variant*

$$S_{\hat{s}-1,j-4^{\gamma_0^c}}^{\gamma_0^c;\hat{h}^{\ell}}=S_{\hat{s},j}^{\gamma_0^c;\hat{h}^{\ell}}=S_{\hat{s}+1,j+4^{\gamma_0^c}}^{\gamma_0^c;\hat{h}^{\ell}}.$$

Proof. We are now commencing the induction process, which will cover γ_0^c. Then we have following cases.

Case $\gamma_0^c=1$:

$$S_{\hat{s},j}^{1;\hat{h}^{\ell}}=\hat{h}_{j-4\hat{s}}^{\ell}=\hat{h}_{j+4-4(\hat{s}+1)}^{\ell}=S_{\hat{s}+1,j+4}^{1;\hat{h}^{\ell}}. \tag{2.13}$$

Similarly

$$S_{\hat{s}+1,j}^{1;\hat{h}^{\ell}}=\hat{h}_{j-4(\hat{s}+1)}^{\ell}=S_{\hat{s},j-4}^{1;\hat{h}^{\ell}}. \tag{2.14}$$

From (2.12), we have

$$S_{\hat{s},j}^{2;\hat{h}^{\ell}}=\sum_{n=4\hat{s}}^{\lfloor j/4\rfloor}S_{\hat{s},n}^{1;\hat{h}^{\ell}}S_{n,j}^{1;\hat{h}^{\ell}}.$$

Using (2.13), we get

$$S_{\hat{s},j}^{2;\hat{h}^{\ell}}=\sum_{n=4\hat{s}}^{\lfloor j/4\rfloor}S_{\hat{s},n}^{1;\hat{h}^{\ell}}S_{n+1,j+4}^{1;\hat{h}^{\ell}}. \tag{2.15}$$

After substituting n by $n-4$ in (2.15), we obtain

$$S_{\hat{s},j}^{2;\hat{h}^{\ell}}=\sum_{n=4(\hat{s}+1)}^{\lfloor j/4+4\rfloor}S_{\hat{s},n-4}^{1;\hat{h}^{\ell}}S_{n-3,j+4}^{1;\hat{h}^{\ell}}.$$

Now by using (2.14), we obtain

$$S_{\hat{s},j}^{2;\hat{h}^{\ell}}=\sum_{n=4(\hat{s}+1)}^{\lfloor j/4+4\rfloor}S_{\hat{s}+1,n}^{1;\hat{h}^{\ell}}S_{n,j+4^2}^{1;\hat{h}^{\ell}}.$$

This implies

$$S_{\hat{s},j}^{2;\hat{h}^{\ell}}=S_{\hat{s}+1,j+4^2}^{2;\hat{h}^{\ell}}.$$

Now, assuming that it is true for an integer $\gamma_0^c=N$, i.e.,

$$S_{\hat{s},j}^{N;\hat{h}^{\ell}}=S_{\hat{s}+1,j+4^N}^{N;\hat{h}^{\ell}}. \tag{2.16}$$

Case $\gamma_0^c=N+1$: Consider

$$S_{\hat{s},j}^{N+1;\hat{h}^{\ell}}=\sum_{n=4\hat{s}}^{\lfloor j/4^N\rfloor}S_{\hat{s},n}^{1;\hat{h}^{\ell}}S_{n,j}^{N;\hat{h}^{\ell}}.$$

By using (2.16), we get

$$S_{\hat{s},j}^{N+1;\hat{h}^{\ell}}=\sum_{n=4\hat{s}}^{\lfloor j/4^N\rfloor}S_{\hat{s},n}^{1;\hat{h}^{\ell}}S_{n+1,j+4^N}^{N;\hat{h}^{\ell}}. \tag{2.17}$$

Now, substituting $n-4$ in place of n in (2.17), we get

$$S^{N+1;\hat{h}^\ell}_{\hat{s},j} = \sum_{n=4(\hat{s}+1)}^{\lfloor j/4^N+4 \rfloor} S^{1;\hat{h}^\ell}_{\hat{s},n-4} S^{N;\hat{h}^\ell}_{n-3,j+4^N}.$$

Using (2.14) and (2.16), we have

$$S^{N+1;\hat{h}^\ell}_{\hat{s},j} = S^{N+1;\hat{h}^\ell}_{\hat{s}+1,j+4^{N+1}}.$$

Similarly, we can prove

$$S^{N+1;\hat{h}^\ell}_{\hat{s},j} = S^{N+1;\hat{h}^\ell}_{\hat{s}-1,j-4^{N+1}}.$$

Hence

$$S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s}-1,j-4^{\gamma_0^c}} = S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s},j} = S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s}+1,j+4^{\gamma_0^c}}.$$

Now the proof is complete. □

Now, we arrive at the following functional result by applying Lemmas 2.1 and 2.2.

Corollary 2.3. *A γ_0^c-times convolution using vector $\hat{h}^\ell = \{\hat{h}^\ell_0, \hat{h}^\ell_1, \dots, \hat{h}^\ell_{4P-1}\}$ has the following associated constants*

$$T_{\gamma_0^c} = \max_j \left\{ \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c} \rfloor} |S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s},j}| \right\} = \max_{j \in \Sigma(\gamma_0^c, P)} \left\{ \sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c} \rfloor} |S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s},j}| \right\}.$$

Proof. Suppose that $\hat{h}^\ell = \{\hat{h}^\ell_0, \hat{h}^\ell_1, \dots, \hat{h}^\ell_{4P-1}\}$, with $P \in \mathbb{N}$ and $\Omega(\gamma_0^c, P) = (4^{\gamma_0^c} - 3)(4P-1)$. Then using Lemma 2.1 and for $j > \Omega(\gamma_0^c, P)$, we obtain

$$S^{\gamma_0^c;\hat{h}^\ell}_{0,j} = 0. \tag{2.18}$$

Similarly, using Lemma 2.2 and for $j > \Omega(\gamma_0^c, P) + \hat{s}4^{\gamma_0^c}$, we get

$$S^{\gamma_0^c;\hat{h}^\ell}_{\hat{s},j} = 0. \tag{2.19}$$

Finally, using (2.18) and (2.19), we get (2.8). □

3. Findings and applications

Now, we discuss some generalized theorems for estimating the error bounds of NSQSS for the curve model followed by determining the subdivision depth based on convolution. Since the proofs of Theorems 3.1 and 3.2 are similar to the one provided in [16], we omit them here.

Theorem 3.1. Consider $\zeta^\ell = \{J^\ell_i; i \in Z, \ell \geqslant 0\}$ be the polygon at ℓ^{th} level of non-stationary subdivision scheme where J^ℓ_i be the points recursively described by (2.1) together with mask condition (2.2), and J^0_i be the points of initial control polygon. Then the error bound between ℓ and $\ell+1$ stages, after two successive iterations, is

$$\|\zeta^{\ell+1} - \zeta^\ell\|_\infty \leqslant \varpi\kappa(T_{\gamma_0^c})^\ell,$$

where $T_{\gamma_0^c}$ for $\gamma_0^c \geqslant 1$ defined in (2.8) and $\kappa = \max_i \|\triangle J^0_i\|$, and

$$\varpi = \max\left(\sum_{\hat{s}=0}^{P-2} |\tilde{c}^\ell_{0,\hat{s}}|, \sum_{\hat{s}=0}^{P-2} |\tilde{c}^\ell_{1,\hat{s}}|, \sum_{\hat{s}=0}^{P-2} |\tilde{c}^\ell_{2,\hat{s}}|, \sum_{\hat{s}=0}^{P-2} |\tilde{c}^\ell_{3,\hat{s}}| \right), \quad \textit{where} \quad \tilde{c}^\ell_{a,\hat{s}} = \sum_{i=\hat{s}+1}^{P-1} c^\ell_{a,i}, \quad 0 \leqslant a \leqslant 3,$$

$$\tilde{c}^\ell_{1,0} = \sum_{i=1}^{P-1} c^\ell_{1,i} - \frac{1}{4}, \quad \tilde{c}^\ell_{2,0} = \sum_{i=1}^{P-1} c^\ell_{2,i} - \frac{2}{4}, \quad \tilde{c}^\ell_{3,0} = \sum_{i=1}^{P-1} c^\ell_{3,i} - \frac{3}{4}.$$

Theorem 3.2. Let ζ^{∞} be the limit curve associated with the subdivision process and ζ^{ℓ} be the curve obtained after ℓ^{th} iterations, then under the similar conditions as in Theorem 3.1, the succeeding result can be demonstrated

$$\rho^{\ell} = \left\|\zeta^{\infty} - \zeta^{\ell}\right\|_{\infty} \leqslant \varpi\kappa\left(\frac{(T_{\gamma_0^c})^{\ell}}{1 - T_{\gamma_0^c}}\right),$$

where $\gamma_0^c \geqslant 1$, such that $T_{\gamma_0^c} < 1$ and $T_{\gamma_0^c}$ is defined in (2.8).

Theorem 3.3. let ρ^{ℓ} be the distance between limit curve ζ^{∞} and its ℓ^{th} level control polygon ζ^{ℓ} and let ℓ be the subdivision depth. For arbitrary error tolerance $\epsilon > 0$, if

$$\ell \geqslant \log_{T_{\gamma_0^c}}\left(\frac{\epsilon(1 - T_{\gamma_0^c})}{\varpi\kappa}\right),$$

then $\rho^{\ell} \leqslant \epsilon$.

Proof. Let, the error bound between limit curve ζ^{∞} and control polygon or subdivision curve ζ^{ℓ} after ℓ^{th} level defined in Theorem 3.2 is ρ^{ℓ}, such that

$$\rho^{\ell} = \left\|\zeta^{\infty} - \zeta^{\ell}\right\|_{\infty} \leqslant \varpi\kappa\left(\frac{(T_{\gamma_0^c})^{\ell}}{1 - T_{\gamma_0^c}}\right).$$

To acquire the given error tolerance $\epsilon > 0$, consider the following

$$\varpi\kappa\left(\frac{(T_{\gamma_0^c})^{\ell}}{1 - T_{\gamma_0^c}}\right) \leqslant \epsilon,$$

implies that

$$\frac{\varpi\kappa}{\epsilon(1 - T_{\gamma_0^c})} \leqslant (T_{\gamma_0^c}^{-1})^{\ell},$$

and further implies that

$$\ell \geqslant \frac{\log\left(\frac{\varpi\kappa}{\epsilon(1 - T_{\gamma_0^c})}\right)}{\log(T_{\gamma_0^c}^{-1})} = \frac{\log\left(\frac{\varpi\kappa}{\epsilon(1 - T_{\gamma_0^c})}\right)}{-\log(T_{\gamma_0^c})} = \log_{T_{\gamma_0^c}}\left(\frac{\varpi\kappa}{\epsilon(1 - T_{\gamma_0^c})}\right)^{-1} = \log_{T_{\gamma_0^c}}\left(\frac{\epsilon(1 - T_{\gamma_0^c})}{\varpi\kappa}\right),$$

then $\rho^{\ell} \leqslant \epsilon$. The proof is now complete. □

3.1. *Numerical experiments for curve case*

Here are a few numerical applications for calculating the subdivision depths of NSQSS for the univariate case described.

Example 3.4. Consider the initial control polygon $J_i^0 = J_i, i \in \mathbb{Z}$ with $J_i^{\ell}, \ell \geqslant 1$ be described recursively by the four-point approximating NSQSS presented in [1]. That is

$$\begin{aligned}
J_{4i}^{\ell+1} &= -\eta_{1,0}^{\ell} J_{i-1}^{\ell} + \eta_{1,1}^{\ell} J_i^{\ell} + \eta_{1,2}^{\ell} J_{i+1}^{\ell} - \eta_{1,3}^{\ell} J_{i+2}^{\ell},\\
J_{4i+1}^{\ell+1} &= -\eta_{2,0}^{\ell} J_{i-1}^{\ell} + \eta_{2,1}^{\ell} J_i^{\ell} + \eta_{2,2}^{\ell} J_{i+1}^{\ell} - \eta_{2,3}^{\ell} J_{i+2}^{\ell},\\
J_{4i+2}^{\ell+1} &= -\eta_{2,3}^{\ell} J_{i-1}^{\ell} + \eta_{2,2}^{\ell} J_i^{\ell} + \eta_{2,1}^{\ell} J_{i+1}^{\ell} - \eta_{2,0}^{\ell} J_{i+2}^{\ell},\\
J_{4i+3}^{\ell+1} &= -\eta_{1,3}^{\ell} J_{i-1}^{\ell} + \eta_{1,2}^{\ell} J_i^{\ell} + \eta_{1,1}^{\ell} J_{i+1}^{\ell} - \eta_{1,0}^{\ell} J_{i+2}^{\ell},
\end{aligned} \tag{3.1}$$

where

$$\eta_{1,0}^{\ell} = \frac{\cos\left(\frac{9\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{7\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{15\nu}{2.4^{\ell+1}}\right)}{\sin\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)\sin\left(\frac{3\nu}{2.4^{\ell}}\right)}, \quad \eta_{1,1}^{\ell} = \frac{\cos\left(\frac{\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{9\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{7\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{15\nu}{2.4^{\ell+1}}\right)}{\sin^2\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)},$$

$$\eta_{1,2}^{\ell}=\frac{\cos\left(\frac{7\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{9\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{15\nu}{2.4^{\ell+1}}\right)}{\sin^2\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)},\quad \eta_{1,3}^{\ell}=\frac{\cos\left(\frac{15\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{7\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{9\nu}{2.4^{\ell+1}}\right)}{\sin\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)\sin\left(\frac{3\nu}{2.4^{\ell}}\right)},$$
$$\eta_{2,0}^{\ell}=\frac{\cos\left(\frac{11\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{3\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{5\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{13\nu}{2.4^{\ell+1}}\right)}{\sin\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)\sin\left(\frac{3\nu}{2.4^{\ell}}\right)},\quad \eta_{2,1}^{\ell}=\frac{\cos\left(\frac{3\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{11\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{5\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{13\nu}{2.4^{\ell+1}}\right)}{\sin^2\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)},$$
$$\eta_{2,2}^{\ell}=\frac{\cos\left(\frac{5\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{11\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{3\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{13\nu}{2.4^{\ell+1}}\right)}{\sin^2\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)},\quad \eta_{2,3}^{\ell}=\frac{\cos\left(\frac{13\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{3\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{5\nu}{4.4^{\ell+1}}\right)\sin\left(\frac{11\nu}{2.4^{\ell+1}}\right)}{\sin\left(\frac{\nu}{2.4^{\ell}}\right)\sin\left(\frac{2\nu}{2.4^{\ell}}\right)\sin\left(\frac{3\nu}{2.4^{\ell}}\right)}.$$

For this quaternary 4-point subdivision scheme $(P=4)$, we have from (2.8),

$$T_{\gamma_0^c}=\max_{j\in\sum(\gamma_0^c,4)}\left\{\sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c}\rfloor}|S_{\hat{s},j}^{\gamma_0^c;\hat{h}^\ell}|\right\}.$$

For $\gamma_0^c=1$, we have

$$T_1=\max_{j\in\sum(1,4)}\left\{\sum_{\hat{s}=0}^{\lfloor j/4^1\rfloor}|S_{\hat{s},j}^{1;\hat{h}^\ell}|\right\}=\max_{j\in\{12,13,14,15\}}\left\{\sum_{\hat{s}=0}^{\lfloor j/4^1\rfloor}|\hat{h}_{j-4\hat{s}}^{\ell}|\right\}.$$

Using Lemma 2.1 and (2.4), we obtain $\hat{h}^\ell=\{\hat{h}_0^\ell,\hat{h}_1^\ell,\hat{h}_2^\ell,\hat{h}_3^\ell,\hat{h}_4^\ell,\hat{h}_5^\ell,\dots,\hat{h}_{14}^\ell,\hat{h}_{15}^\ell\}$ with $\hat{h}_n^\ell=0$ for $n\geqslant 16$. Now consider

$$T_1=\max\left\{\sum_{\hat{s}=0}^{\lfloor 12/4\rfloor}|\hat{h}_{12-4\hat{s}}^{\ell}|,\sum_{\hat{s}=0}^{\lfloor 13/4\rfloor}|\hat{h}_{13-4\hat{s}}^{\ell}|,\sum_{\hat{s}=0}^{\lfloor 14/4\rfloor}|\hat{h}_{14-4\hat{s}}^{\ell}|,\sum_{\hat{s}=0}^{\lfloor 15/4\rfloor}|\hat{h}_{15-4\hat{s}}^{\ell}|\right\}.$$

This implies that

$$\begin{aligned}T_1&=\max\left\{|\hat{h}_{12}^\ell|+|\hat{h}_8^\ell|+|\hat{h}_4^\ell|+|\hat{h}_0^\ell|,|\hat{h}_{13}^\ell|+|\hat{h}_9^\ell|+|\hat{h}_5^\ell|+|\hat{h}_1^\ell|,|\hat{h}_{14}^\ell|+|\hat{h}_{10}^\ell|+|\hat{h}_6^\ell|+|\hat{h}_2^\ell|,|\hat{h}_{15}^\ell|+|\hat{h}_{11}^\ell|+|\hat{h}_7^\ell|+|\hat{h}_3^\ell|\right\}\\&=\max\left\{0.3173696217,0.2905725894,0.3173696218,0.3323767041\right\}=0.3323767041.\end{aligned}$$

Similarly, we can calculate the values of $T_{\gamma_0^c},\gamma_0^c\geqslant 2$. For ease, we have calculated the values up to $\gamma_0^c=5$.

$$\begin{cases}\bullet\ \text{Convolution coefficients for } \ell=1,\ T_1=0.3323767041,\ T_2=0.1067695669,\\ \quad T_3=0.0348669469,\ T_4=0.0112772724,\ \text{and } T_5=0.0036495886.\\ \bullet\ \text{Convolution coefficients for } \ell=2,\ T_1=0.3320519013,\ T_2=0.1067908939,\\ \quad T_3=0.0347004408,\ T_4=0.0112015818,\ \text{and } T_5=0.0036181802.\\ \bullet\ \text{Convolution coefficients for } \ell=3,\ T_1=0.3320325371,\ T_2=0.1067708929,\\ \quad T_3=0.0346903059,\ T_4=0.0111969710,\ \text{and } T_5=0.0036162672.\\ \bullet\ \text{Convolution coefficients for } \ell=4,\ T_1=0.3320313304,\ T_2=0.1067696450,\\ \quad T_3=0.0346896735,\ T_4=0.0111966833,\ \text{and } T_5=0.0036161478.\\ \bullet\ \text{Convolution coefficients for } \ell=5,\ T_1=0.3320312550,\ T_2=0.10676956697,\\ \quad T_3=0.0346896339,\ T_4=0.0111966653,\ \text{and } T_5=0.0036161404.\end{cases}\tag{3.2}$$

Remark 3.5. Here, $T_{\gamma_0^c}$ for $\gamma_0^c=1$ is equal to δ_1 described in [16]. From (3.2), we analyze that as we increase the order of convolution γ_0^c, the value of $T_{\gamma_0^c}$ decreases. When the value of $T_{\gamma_0^c}$ decreases, we obtain fewer iterations (subdivision depth) compared to the previous ones, as verified from Table 1. Note that there was a strong condition in [16] that if $\delta_1 > 1$, it is impossible to calculate error bounds. However, using the proposed technique, one can calculate the error bounds for those NSQSS with $T_{\gamma_0^c} \geqslant 1$. Additionally, using the proposed technique, less computational cost is consumed. Therefore, these are the main advantages of the proposed approach.

After the computation of convolution coefficients for the proposed scheme and using Theorem 3.3, the computation of subdivision depth at different levels ℓ, is illustrated in Table 1.

Table 1: Subdivision depth of 4-point approximating NSQSS.

ϵ	$5.49e^{-9}$					$7.31e^{-14}$				
$T_{\gamma_0^c}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
T_1	16	16	16	16	16	26	26	26	26	26
T_2	8	8	8	8	8	13	13	13	13	13
T_3	5	5	5	5	5	8	8	8	8	8
T_4	4	4	4	4	4	6	6	6	6	6
T_5	3	3	3	3	3	5	5	5	5	5
ϵ	$2.67e^{-16}$					$9.74e^{-19}$				
$T_{\gamma_0^c}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
T_1	31	31	31	31	31	36	36	36	36	36
T_2	15	15	15	15	15	18	18	18	18	18
T_3	10	10	10	10	10	12	12	12	12	12
T_4	8	7	7	7	7	9	9	9	9	9
T_5	6	6	6	6	6	7	7	7	7	7

Remark 3.6. In (3.2) and Table 1, for $\ell = 1$ to $\ell = 5$, the different behavior of the non-stationary scheme given in Example 3.4 is shown. In Table 1, the rows of T_1, T_2, T_3, T_4, and T_5 show different subdivision depths (number of iterations to get the desired model). For instance, obtaining an error tolerance of $7.31e^{-14}$ requires twenty-six iterations by the method given in [16], but with our method, it only requires five iterations, corresponding to T_5. The graphical representation or comparison of these convolution results is presented in Figure 1 (a).

Example 3.7. Consider the 4-point interpolating NSQSS presented in [28]:

$$\begin{aligned}
J_{4i}^{\ell+1} &= J_i^{\ell},\\
J_{4i+1}^{\ell+1} &= -a_0^{\ell}J_{i-1}^{\ell} + a_1^{\ell}J_i^{\ell} + a_2^{\ell}J_{i+1}^{\ell} - a_3^{\ell}J_{i+2}^{\ell},\\
J_{4i+2}^{\ell+1} &= -b_0^{\ell}J_{i-1}^{\ell} + b_1^{\ell}J_i^{\ell} + b_2^{\ell}J_{i+1}^{\ell} - b_3^{\ell}J_{i+2}^{\ell},\\
J_{4i+2}^{\ell+1} &= -a_3^{\ell}J_{i-1}^{\ell} + a_2^{\ell}J_i^{\ell} + a_1^{\ell}J_{i+1}^{\ell} - a_0^{\ell}J_{i+2}^{\ell},
\end{aligned} \tag{3.3}$$

where

$$a_0^{\ell} = \frac{\sin\left(\frac{\mu}{2^{\ell+4}}\right)\sin\left(\frac{3\mu}{2^{\ell+4}}\right)\sin\left(\frac{7\mu}{2^{\ell+4}}\right)}{\sin\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)\sin\left(\frac{3\mu}{2^{\ell+2}}\right)}, \qquad a_1^{\ell} = \frac{\sin\left(\frac{3\mu}{2^{\ell+4}}\right)\sin\left(\frac{5\mu}{2^{\ell+4}}\right)\sin\left(\frac{7\mu}{2^{\ell+4}}\right)}{\sin^2\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)},$$

$$a_2^{\ell} = \frac{\sin\left(\frac{\mu}{2^{\ell+4}}\right)\sin\left(\frac{5\mu}{2^{\ell+4}}\right)\sin\left(\frac{7\mu}{2^{\ell+4}}\right)}{\sin^2\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)}, \qquad a_3^{\ell} = \frac{\sin\left(\frac{\mu}{2^{\ell+4}}\right)\sin\left(\frac{3\mu}{2^{\ell+4}}\right)\sin\left(\frac{5\mu}{2^{\ell+4}}\right)}{\sin\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)\sin\left(\frac{3\mu}{2^{\ell+2}}\right)},$$

$$b_0^{\ell} = b_3^{\ell} = \frac{\sin^2\left(\frac{\mu}{2^{\ell+3}}\right)\sin\left(\frac{3\mu}{2^{\ell+3}}\right)}{\sin\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)\sin\left(\frac{3\mu}{2^{\ell+2}}\right)}, \qquad b_1^{\ell} = b_2^{\ell} = \frac{\sin\left(\frac{\mu}{2^{\ell+3}}\right)\sin^2\left(\frac{3\mu}{2^{\ell+3}}\right)}{\sin^2\left(\frac{\mu}{2^{\ell+2}}\right)\sin\left(\frac{2\mu}{2^{\ell+2}}\right)}.$$

Now apply the convolution to find $T_{\gamma_0^c}$ for $\gamma_0^c \geqslant 1$ defined in (2.8) for the proposed scheme at different subdivision levels are shown in (3.4).

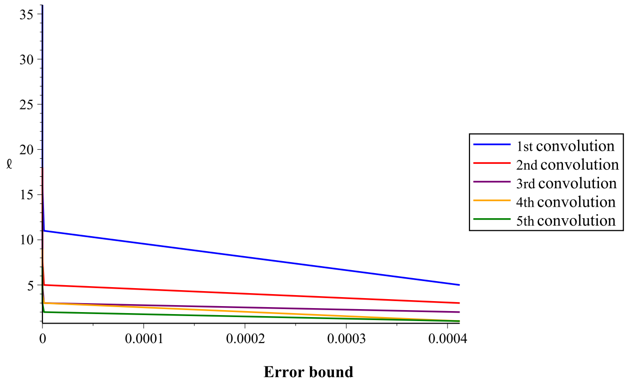
$$\begin{cases} \bullet \text{ Convolution coefficients for } \ell = 1,\ T_1 = 0.3308434393,\ T_2 = 0.1081304002, \\ \quad T_3 = 0.0352273549,\ T_4 = 0.0114818856, \text{ and } T_5 = 0.0037421898. \\ \bullet \text{ Convolution coefficients for } \ell = 2,\ T_1 = 0.3287889801,\ T_2 = 0.1068579542, \\ \quad T_3 = 0.03462038451,\ T_4 = 0.01122113427, \text{ and } T_5 = 0.0036367912. \\ \bullet \text{ Convolution coefficients for } \ell = 3,\ T_1 = 0.3282900352,\ T_2 = 0.1065479452, \\ \quad T_3 = 0.03447270344,\ T_4 = 0.0111579122, \text{ and } T_5 = 0.0036113316. \\ \bullet \text{ Convolution coefficients for } \ell = 4,\ T_1 = 0.3281661991,\ T_2 = 0.1064709379, \\ \quad T_3 = 0.0344360306,\ T_4 = 0.0111422266, \text{ and } T_5 = 0.00360502079. \\ \bullet \text{ Convolution coefficients for } \ell = 5,\ T_1 = 0.3281352960,\ T_2 = 0.1064517168, \\ \quad T_3 = 0.0344268777,\ T_4 = 0.0111383127, \text{ and } T_5 = 0.0036034464. \end{cases} \tag{3.4}$$

If the convolution coefficient $T_{\gamma_0^c}$ is greater than or equal to one then we have to apply the γ_0^c-times convolution unit $T_{\gamma_0^c}$ becomes less than one. To obtain a smaller value of $T_{\gamma_0^c}$ convolution may also be applied even $T_{\gamma_0^c} < 1$. A smaller value of $T_{\gamma_0^c}$ produces better outcomes. From the above computations (3.4), we can observe that, with the increase in the order of convolution γ_0^c the value of $T_{\gamma_0^c}$ decreases. For example, for $\ell = 1$ and $\gamma_0^c = 1$ the value of $T_{\gamma_0^c} = 0.3308434393$ but for $\gamma_0^c = 5$ the value of $T_{\gamma_0^c} = 0.0037421898$. $T_{\gamma_0^c}$ are calculated up to fifth convolution. After the estimation of convolution coefficients for the proposed scheme and using Theorem 3.3 the computation of subdivision depth at different levels ℓ, are illustrated in Table 2.

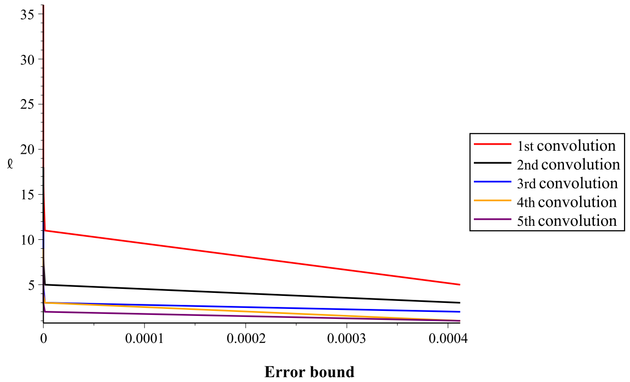
Table 2: Subdivision depth of 4-point interpolating NSQSS.

ϵ	$1.97e^{-11}$					$7.37e^{-14}$				
$T_{\gamma_0^c}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
T_1	21	20	20	20	20	26	25	25	25	25
T_2	10	10	10	10	10	13	13	13	13	13
T_3	7	7	7	7	7	8	8	8	8	8
T_4	5	5	5	5	5	6	6	6	6	6
T_5	4	4	4	4	4	5	5	5	5	5
ϵ	$2.76e^{-16}$					$1.03e^{-18}$				
$T_{\gamma_0^c}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
T_1	31	30	30	30	30	36	36	35	35	35
T_2	15	15	15	15	15	18	18	18	18	18
T_3	10	10	10	10	10	12	12	12	12	12
T_4	8	7	7	7	7	9	9	9	9	9
T_5	6	6	6	6	6	7	7	7	7	7

Remark 3.8. In Table 2, the computation of subdivision depth at different levels ℓ correlated with the predefined error tolerance is presented. For example, twenty-six iterations are required to attain a given error tolerance of $7.37e^{-14}$ by the process given in [16], but with our approach, only five iterations corresponding to T_5 are needed. The graphical comparison of different convolutions is demonstrated in Figure 1 (b).



(a) 4-point approximating NSQSS [1].



(b) 4-point interpolating NSQSS [28].

Figure 1: Figures (a) and (b) represents the comparison between different convolution results for the curve case. This shows that the error decreases with the increase of order of convolution. Here horizontal axis shows the error bound and vertical axis shows the subdivision level ℓ.

4. Non-stationary quaternary bivariate case

Let the points $\{J^{\ell}_{i,j}; i,j \in \mathbb{Z}\}$ represents a sequence in $\mathbb{R}^{\mathbb{N}}$, $\mathbb{N} \geqslant 2$ for the ℓ-th level surface case and is described as

$$J^{\ell+1}_{4i+\mu,4j+\nu} = \sum_{\hat{r}=0}^{P-1}\sum_{\hat{a}=0}^{P-1} c^{\ell}_{\mu,\hat{r}} c^{\ell}_{\nu,\hat{a}} J^{\ell}_{i+\hat{r},j+\hat{a}}, \quad \mu,\nu = 0,1,2,3, \tag{4.1}$$

where $c^{\ell}_{\mu,\hat{r}}$ and $c^{\ell}_{\nu,\hat{a}}$ satisfies (2.2). Now, consider the new expressions for $\hat{r},\hat{a} = 0,1,\dots,P-1$ and assign the coefficients $E^{\ell} = \{E^{\ell}_{n}\}_{n\in\mathbb{N}}$ and $F^{\ell} = \{F^{\ell}_{n}\}_{n\in\mathbb{N}}$ such that

$$\begin{cases} E^{\ell}_{4\hat{r}+\iota} = c^{\ell}_{\iota,P-\hat{r}-1}, & \iota = 0,1,2,3, \text{ and } \hat{r} = 0,\dots,P-1,\\ F^{\ell}_{4\hat{a}+\varphi} = q^{\ell}_{\varphi,P-\hat{a}-1}, & \varphi = 0,1,2,3, \text{ and } \hat{a} = 0,\dots,P-1. \end{cases} \tag{4.2}$$

4.1. *Convolution and subdivision depth for quaternary non-stationary bivariate case*

In this section, we first present a few notations and the main results of convolutions for the surface model, then describe the results for error bounds and subdivision depths of non-stationary quaternary subdivision surfaces (NSQSS).

Lemma 4.1. *Let, a two dimensional vector* $\{J^{\ell}_{m,n}; m,n \geqslant 0\}$ *and* $\{E^{\ell}_{n}; n \geqslant 0\} = (E^{\ell}_{n})^{4P-1}_{n=0}$, $\{F^{\ell}_{n}; n \geqslant 0\} = (F^{\ell}_{n})^{4P-1}_{n=0}$ *with* $E^{\ell}_{n} = F^{\ell}_{n} = 0$ *for* $n \geqslant 4P$. *Then convolution of* $J^{\ell} = J^{\ell}_{n}$, $E^{\ell} = E^{\ell}_{n}$, *and* $F^{\ell} = F^{\ell}_{n}$ *for NSQSS is given by*

$$J^{\gamma^s_0;\ell}_{i,j} = \left(J^{\gamma^s_0-1;0;\ell} \star E^{\ell}F^{\ell}\right)_{i,j} = \sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor} J^{\gamma^s_0-1;\ell}_{m,n} E^{\ell}_{i-4m} F^{\ell}_{j-4n}.$$

Similarly, the γ^s_0*-times convolution reformulations are described as*

$$J^{\gamma^s_0;\ell}_{i,j} = (\cdots(((J^{\gamma^s_0-1;0} \star E^{\ell}F^{\ell}) \star E^{\ell}F^{\ell}) \star \cdots \star E^{\ell}F^{\ell}) \star E^{\ell}F^{\ell})_{i,j} = \sum_{m=0}^{\lfloor i/4^{\gamma^s_0}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma^s_0}\rfloor} J^{0;\ell}_{m,n} S^{\gamma^s_0,E^{\ell}}_{m,i} S^{\gamma^s_0,F^{\ell}}_{n,j}, \tag{4.3}$$

with

$$\begin{cases} S^{1;E^{\ell}}_{m,i} = E^{\ell}_{i-4m}, \text{ and } S^{\gamma^s_0;E^{\ell}}_{m,i} = \displaystyle\sum_{J=4m}^{\lfloor i/4^{\gamma^s_0-1}\rfloor} S^{1;E^{\ell}}_{m,J} S^{\gamma^s_0-1;E^{\ell}}_{J,i},\\ S^{1;F^{\ell}}_{n,j} = F^{\ell}_{j-4n}, \text{ and } S^{\gamma^s_0;F^{\ell}}_{n,j} = \displaystyle\sum_{r=4n}^{\lfloor j/4^{\gamma^s_0-1}\rfloor} S^{1;F^{\ell}}_{n,r} S^{\gamma^s_0-1;F^{\ell}}_{r,j}, \ \gamma^s_0 \geqslant 2. \end{cases} \tag{4.4}$$

From (4.3), we have

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}| \leqslant M_{\gamma_0^s}N_{\gamma_0^s}\max_{m,n}|J_{m,n}^0|.$$

Here

$$M_{\gamma_0^s} = \max_i\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}|\right\}, \tag{4.5}$$

and

$$N_{\gamma_0^s} = \max_j\left\{\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{n,j}^{\gamma_0^s,F^\ell}|\right\}. \tag{4.6}$$

Also

$$\max_{i,j}\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}||S_{n,j}^{\gamma_0^s,F^\ell}|\right\} = \max_{i,j\in\Sigma(\gamma_0^s,P)}\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}||S_{n,j}^{\gamma_0^s,F^\ell}|\right\}, \tag{4.7}$$

where $\Sigma(\gamma_0^s, P)$ is defined in (2.9).

Proof. To prove this result, we start with the case $\gamma_0^s = 1$ and $\gamma_0^s = 2$ convolution and afterward we examine the general case.

Case $\gamma_0^s = 1$: Let $J_{i,j}$ be an arbitrary sequence of vectors, then we have

$$J_{i,j}^{\gamma_0^s;\ell} = \left(J^{\gamma_0^s-1;0;\ell}\star E^\ell F^\ell\right)_{i,j} = \sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}J_{m,n}^{\gamma_0^s-1;\ell}E_{i-4m}^\ell F_{j-4n}^\ell.$$

Where we suppose $E_{i-4m}^\ell = S_{m,i}^{\gamma_0^s,E^\ell}$ and $F_{j-4n}^\ell = S_{n,j}^{\gamma_0^s,F^\ell}$ for arbitrary sequences E^ℓ and F^ℓ, thus

$$J_{i,j}^{\gamma_0^s;\ell} = \left(J^{\gamma_0^s-1;0;\ell}\star E^\ell F^\ell\right)_{i,j} = \sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}J_{m,n}^{\gamma_0^s-1;\ell}S_{m,i}^{1,E^\ell}S_{n,j}^{1,F^\ell},$$

implies that

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}| = \max_{i,j}\left|\sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}J_{m,n}^{\gamma_0^s-1;\ell}S_{m,i}^{1,E^\ell}S_{n,j}^{1,F^\ell}\right| \leqslant \max_{i,j}|\sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}|S_{m,i}^{1,E^\ell}||S_{n,j}^{1,F^\ell}|\max_{m,n}||J_{m,n}^{\gamma_0^s-1;\ell}|. \tag{4.8}$$

Consider

$$M_1 = \max_i\left\{\sum_{m=0}^{\lfloor i/4\rfloor}S_{m,i}^{1,E^\ell}\right\} \quad \text{and} \quad N_1 = \max_j\left\{\sum_{n=0}^{\lfloor j/4\rfloor}S_{n,j}^{1,F^\ell}\right\}.$$

From (4.8), we have

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}| \leqslant M_1N_1\max_{m,n}|J_{m,n}^{\gamma_0^s-1;\ell}|.$$

Case $\gamma_0^s = 2$: Now, by applying two times convolution, we get

$$J_{m,n}^{\gamma_0^s-1;\ell} = \left(J^{\gamma_0^s-2;0;\ell}\star E^\ell F^\ell\right)_{m,n} = \left((J^{\gamma_0^s-1;0;\ell}\star E^\ell F^\ell)\star E^\ell F^\ell\right)_{i,j}.$$

This implies that

$$J_{i,j}^{\gamma_0^s-1;\ell} = \sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}(J^{\gamma_0^s-1;0;\ell}\star E^\ell F^\ell)_{i,j}E_{i-4m}^\ell F_{j-4n}^\ell.$$

This gives

$$J_{i,j}^{\gamma_0^s-1;\ell}=\sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}\left(\sum_{u=0}^{\lfloor m/4\rfloor}\sum_{v=0}^{\lfloor n/4\rfloor}(J_{u,v}^{\gamma_0^s-2;0;\ell}E_{m-4u}^{\ell}F_{n-4v}^{\ell})\right)E_{i-4m}^{\ell}F_{j-4n}^{\ell}.$$

This further implies that

$$J_{i,j}^{\gamma_0^s-1;\ell}=\sum_{m=0}^{\lfloor i/4^2\rfloor}\sum_{n=0}^{\lfloor j/4^2\rfloor}J_{m,n}^{\gamma_0^s-2;\ell}\sum_{w=4m}^{\lfloor i/4\rfloor}E_{w-4m}^{\ell}E_{i-4w}^{\ell}\sum_{x=4n}^{\lfloor j/4\rfloor}F_{x-4n}^{\ell}F_{j-4x}^{\ell}.$$

Again implies

$$J_{i,j}^{\gamma_0^s-1;\ell}=\sum_{m=0}^{\lfloor i/4^2\rfloor}\sum_{n=0}^{\lfloor j/4^2\rfloor}J_{m,n}^{\gamma_0^s-2;\ell}\sum_{w=4m}^{\lfloor i/4\rfloor}S_{m,w}^{1,E^{\ell}}S_{w,i}^{1,E^{\ell}}\sum_{x=4n}^{\lfloor j/4\rfloor}S_{n,x}^{1,F^{\ell}}S_{x,j}^{1,F^{\ell}}.$$

Furthermore

$$J_{i,j}^{\gamma_0^s;\ell}=\sum_{m=0}^{\lfloor i/4^2\rfloor}\sum_{n=0}^{\lfloor j/4^2\rfloor}J_{m,n}^{\gamma_0^s-2;\ell}S_{m,i}^{2,E^{\ell}}S_{n,j}^{2,F^{\ell}}.$$

Now, we get

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}|=\max_{i,j}\left|\sum_{m=0}^{\lfloor i/4^2\rfloor}\sum_{n=0}^{\lfloor j/4^2\rfloor}J_{m,n}^{\gamma_0^s-2;\ell}S_{m,i}^{2,E^{\ell}}S_{n,j}^{2,F^{\ell}}\right|\leqslant\max_{i,j}\sum_{m=0}^{\lfloor i/4^2\rfloor}\sum_{n=0}^{\lfloor j/4^2\rfloor}\left|S_{m,i}^{2,E^{\ell}}\right|\left|S_{n,j}^{2,F^{\ell}}\right|\max_{m,n}\left|J_{m,n}^{\gamma_0^s-2;\ell}\right|. \tag{4.9}$$

Let

$$M_2=\max_i\left\{\sum_{m=0}^{\lfloor i/4^2\rfloor}\left|S_{m,i}^{2,E^{\ell}}\right|\right\}\quad\text{and}\quad N_2=\max_j\left\{\sum_{n=0}^{\lfloor j/4^2\rfloor}\left|S_{n,j}^{2,F^{\ell}}\right|\right\},$$

then, we acquire from (4.9),

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}|\leqslant M_2N_2\max_{m,n}\left|J_{m,n}^{\gamma_0^s-2;\ell}\right|.$$

By applying the same procedure, we get the following reformulations for γ_0^s-th convolution

$$J_{i,j}^{\gamma_0^s;\ell}=\left(J^{\gamma_0^s-\gamma_0^s;0;\ell}\star E^{\ell}F^{\ell}\right)_{m,n}=(\cdots(((J^{\gamma_0^s-1;0;\ell}\star E^{\ell}F^{\ell})\star E^{\ell}F^{\ell})\star\cdots\star E^{\ell}F^{\ell})\star E^{\ell}F^{\ell})_{i,j}.$$

This implies

$$J_{i,j}^{\gamma_0^s;\ell}=\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}J_{m,n}^{0;0;\ell}S_{m,i}^{\gamma_0^s,E^{\ell}}S_{n,j}^{\gamma_0^s,F^{\ell}}=\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}J_{m,n}^{0;\ell}S_{m,i}^{\gamma_0^s,E^{\ell}}S_{n,j}^{\gamma_0^s,F^{\ell}},$$

where

$$S_{m,i}^{\gamma_0^s,E^{\ell}}=\sum_{u=4m}^{\lfloor i/4^{\gamma_0^s-1}\rfloor}S_{m,u}^{\gamma_0^s-1,E^{\ell}}S_{u,i}^{\gamma_0^s-1,E^{\ell}},$$

and

$$S_{n,j}^{\gamma_0^s,F^{\ell}}=\sum_{v=4n}^{\lfloor j/4^{\gamma_0^s-1}\rfloor}S_{n,v}^{\gamma_0^s-1,F^{\ell}}S_{n,j}^{\gamma_0^s-1,F^{\ell}}.$$

Now

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}| = \max_{i,j}\left|\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor} J_{m,n}^{0;\ell} S_{m,i}^{\gamma_0^s,E^\ell} S_{n,j}^{\gamma_0^s,F^\ell}\right| \leqslant \max_{i,j}\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}\left|S_{m,i}^{\gamma_0^s,E^\ell}\right|\left|S_{n,j}^{\gamma_0^s,F^\ell}\right|\max_{m,n}\left|J_{m,n}^{0;\ell}\right|. \quad (4.10)$$

Now consider the following

$$M_{\gamma_0^s} = \max_i\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}|\right\} = \max_{i\in\sum(\gamma_0^s,P)}\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}|\right\},$$

and

$$N_{\gamma_0^s} = \max_j\left\{\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{n,j}^{\gamma_0^s,F^\ell}|\right\} = \max_{j\in\sum(\gamma_0^s,P)}\left\{\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{n,j}^{\gamma_0^s,F^\ell}|\right\}.$$

then, from (4.10), we have

$$\max_{i,j}|J_{i,j}^{\gamma_0^s;\ell}| \leqslant M_{\gamma_0^s}N_{\gamma_0^s}\max_{m,n}|J_{m,n}^0|,$$

where

$$\max_{i,j}\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}||S_{n,j}^{\gamma_0^s,F^\ell}|\right\} = \max_{i,j\in\Sigma(\gamma_0^s,P)}\left\{\sum_{m=0}^{\lfloor i/4^{\gamma_0^s}\rfloor}\sum_{n=0}^{\lfloor j/4^{\gamma_0^s}\rfloor}|S_{m,i}^{\gamma_0^s,E^\ell}||S_{n,j}^{\gamma_0^s,F^\ell}|\right\},$$

where $\Sigma(\gamma_0^s, P)$ is defined in (2.9). This completes the proof. □

Now, the generalized results are presented for finding the error bounds of NSQSS, followed by an improved subdivision depth computation technique based on these error bounds. We omit the proof of Theorems 4.2 and 4.3 since it is similar to the one given in [16].

Theorem 4.2. Let $\zeta^\ell = \{J_{i,j}^\ell; i,j \in \mathbb{Z}, \ell \geqslant 0\}$ the the polygon at the ℓ^{th} level of NSQSS, where $J_{i,j}^\ell$ be the points recursively described in (4.1) along with the condition (2.2). Also let $\{J_{i,j}^0, i,j \in \mathbb{Z}\}$ to be the first control polygon. Then the error bounds of two successive refinements between the level ℓ and $\ell+1$, using the similar technique given in [16], is

$$\|\zeta^{\ell+1} - \zeta^\ell\|_\infty \leqslant \vartheta(M_{\gamma_0^s}N_{\gamma_0^s})^\ell,$$

where $M_{\gamma_0^s}, N_{\gamma_0^s}, \gamma_0^s \geqslant 1$ defined in (4.5) and (4.6), and $\vartheta = \max_{\alpha,\beta}\left\{\sum_{t=1}^3(\chi_t)(\eta_{\alpha,\beta}^t), \alpha,\beta = 0,1,2,3\right\}$, where χ_t and $\eta_{\alpha,\beta}^t$ for $\alpha,\beta = 0,1,2,3$ are defined in [16].

Theorem 4.3. Under the same circumstances used in Theorem 4.2, let ζ^∞ be the limit surface associated with the subdivision process. Then

$$\hbar^\ell = \|\zeta^\infty - \zeta^\ell\|_\infty \leqslant \vartheta\left(\frac{(M_{\gamma_0^s}N_{\gamma_0^s})^\ell}{1 - M_{\gamma_0^s}N_{\gamma_0^s}}\right),$$

where $\gamma_0^s \geqslant 1$ is a natural number, such that $M_{\gamma_0^s}N_{\gamma_0^s} < 1$.

Theorem 4.4. Let ℓ be the subdivision depth and let $\hbar^\ell$ be the error bound between NSQSS ζ^∞ and its ℓ^{th} level control polygon ζ^ℓ. For arbitrary $\epsilon > 0$, if

$$\ell \geqslant \log_{(M_{\gamma_0^s}G_{\gamma_0^s})}\left(\frac{\epsilon(1 - M_{\gamma_0^s}N_{\gamma_0^s})}{\vartheta}\right),$$

then $\hbar^\ell \leqslant \epsilon$.

Proof. The proof in Theorem 3.3 is resemblant. □

4.2. Numerical experiments for bivariate case

Here, some numerical examples to estimate error bounds and subdivision depth for surface models are presented. First, we find the term $M_{\gamma_0^s}N_{\gamma_0^s}$ for $\gamma_0^s \geqslant 1$ using (4.5) and (4.6). From the calculated results, we observe that the value of $M_{\gamma_0^s}N_{\gamma_0^s}$ decreases with the increase in the order of convolution γ_0^s. This is the main advantage of our work. Graphical representation is also presented in Figure 2.

Example 4.5. Consider the tensor product of 4-point approximating NSQSS given in (3.1). Now apply four-times convolution γ_0^s to find the four convolution coefficients $M_{\gamma_0^s}N_{\gamma_0^s}$ defined by (4.5) and (4.6).

First convolution (i.e., $\gamma_0^s = 1$): From (2.9) and (2.10), we get $\Omega(1,4) = 15$ and $\Sigma(1,4) = \{12, 13, 14, 15\}$. Now from (4.7), we get

$$
\begin{aligned}
M_1N_1 &= \max_{i,j\in\{12,13,14,15\}}\left\{\sum_{m=0}^{\lfloor i/4\rfloor}\sum_{n=0}^{\lfloor j/4\rfloor}|S^{1,E^\ell}_{m,i}||S^{1,F^\ell}_{n,j}|\right\}\\
&= \max\left\{\sum_{m=0}^{\lfloor 12/4\rfloor}\sum_{n=0}^{\lfloor 12/4\rfloor}|S^{1,E^\ell}_{m,12}||S^{1,F^\ell}_{n,12}|, \sum_{m=0}^{\lfloor 12/4\rfloor}\sum_{n=0}^{\lfloor 13/4\rfloor}|S^{1,E^\ell}_{m,12}||S^{1,F^\ell}_{n,13}|, \sum_{m=0}^{\lfloor 12/4\rfloor}\sum_{n=0}^{\lfloor 14/4\rfloor}|S^{1,E^\ell}_{m,12}||S^{1,F^\ell}_{n,14}|,\right.\\
&\quad \sum_{m=0}^{\lfloor 12/4\rfloor}\sum_{n=0}^{\lfloor 15/4\rfloor}|S^{1,E^\ell}_{m,12}||S^{1,F^\ell}_{n,15}|, \sum_{m=0}^{\lfloor 13/4\rfloor}\sum_{n=0}^{\lfloor 12/4\rfloor}|S^{1,E^\ell}_{m,13}||S^{1,F^\ell}_{n,12}|, \sum_{m=0}^{\lfloor 13/4\rfloor}\sum_{n=0}^{\lfloor 13/4\rfloor}|S^{1,E^\ell}_{m,13}||S^{1,F^\ell}_{n,13}|,\\
&\quad \sum_{m=0}^{\lfloor 13/4\rfloor}\sum_{n=0}^{\lfloor 14/4\rfloor}|S^{1,E^\ell}_{m,13}||S^{1,F^\ell}_{n,14}|, \sum_{m=0}^{\lfloor 13/4\rfloor}\sum_{n=0}^{\lfloor 15/4\rfloor}|S^{1,E^\ell}_{m,13}||S^{1,F^\ell}_{n,15}|, \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 12/4\rfloor}|S^{1,E^\ell}_{m,14}||S^{1,F^\ell}_{n,12}|,\\
&\quad \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 13/4\rfloor}|S^{1,E^\ell}_{m,14}||S^{1,F^\ell}_{n,13}|, \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 14/4\rfloor}|S^{1,E^\ell}_{m,14}||S^{1,F^\ell}_{n,14}|, \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 15/4\rfloor}|S^{1,E^\ell}_{m,14}||S^{1,F^\ell}_{n,15}|,\\
&\quad \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 12/4\rfloor}|S^{1,E^\ell}_{m,15}||S^{1,F^\ell}_{n,12}|, \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 13/4\rfloor}|S^{1,E^\ell}_{m,15}||S^{1,F^\ell}_{n,13}|, \sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 14/4\rfloor}|S^{1,E^\ell}_{m,15}||S^{1,F^\ell}_{n,14}|,\\
&\quad \left.\sum_{m=0}^{\lfloor 14/4\rfloor}\sum_{n=0}^{\lfloor 15/4\rfloor}|S^{1,E^\ell}_{m,15}||S^{1,F^\ell}_{n,15}|\right\}.
\end{aligned}
$$

Now from (4.4), we have

$$
\begin{aligned}
M_1N_1 = \max&\left\{\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{12-4m}||F^\ell_{12-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{12-4m}||F^\ell_{13-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{12-4m}||F^\ell_{14-4n}|,\right.\\
&\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{12-4m}||F^\ell_{15-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{13-4m}||F^\ell_{12-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{13-4m}||F^\ell_{13-4n}|,\\
&\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{13-4m}||F^\ell_{14-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{13-4m}||F^\ell_{15-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{14-4m}||F^\ell_{12-4n}|,\\
&\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{14-4m}||F^\ell_{13-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{14-4m}||F^\ell_{14-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{14-4m}||F^\ell_{15-4n}|,\\
&\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{15-4m}||F^\ell_{12-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{15-4m}||F^\ell_{13-4n}|, \sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{15-4m}||F^\ell_{14-4n}|,\\
&\left.\sum_{m=0}^{1}\sum_{n=0}^{1}|E^\ell_{15-4m}||F^\ell_{15-4n}|\right\}.
\end{aligned}
$$

This implies

$$M_1N_1 = \max\{0.3525036074, 0.3927105546, 0.3927105546, 0.3525036074, 0.3227400449, 0.3595521276, 0.3595521276, 0.3227400449, 0.3525036070, 0.3927105545, 0.3927105545, 0.3525036070, 0.3691720288, 0.4112801945, 0.4112801945, 0.3691720288\} = 0.4112801945.$$

The values of the coefficients $M_{\gamma_0^s}N_{\gamma_0^s}$ are displayed numerically in (4.11). Using Theorem 4.4, the number of iterations (subdivision depth) for different levels of iteration ℓ is presented in Table 3.

$$\begin{cases} \bullet \text{ Convolution coefficients for } \ell = 1,\ M_1N_1 = 0.4112801945,\ M_2N_2 = 0.1004142118, \\ \quad M_3N_3 = 0.02497913514, \text{ and } M_4N_4 = 0.006236920019. \\ \bullet \text{ Convolution coefficients for } \ell = 2,\ M_1N_1 = 0.4099386719,\ M_2N_2 = 0.09935220356, \\ \quad M_3N_3 = 0.02471964933, \text{ and } M_4N_4 = 0.006170894633. \\ \bullet \text{ Convolution coefficients for } \ell = 3,\ M_1N_1 = 0.4098565426,\ M_2N_2 = 0.09909265225, \\ \quad M_3N_3 = 0.02465535416, \text{ and } M_4N_4 = 0.006154699441. \\ \bullet \text{ Convolution coefficients for } \ell = 4,\ M_1N_1 = 0.4098514156,\ M_2N_2 = 0.09902812883, \\ \quad M_3N_3 = 0.02463931622, \text{ and } M_4N_4 = 0.006150669774. \\ \bullet \text{ Convolution coefficients for } \ell = 5,\ M_1N_1 = 0.4098510948,\ M_2N_2 = 0.09901202020, \\ \quad M_3N_3 = 0.02463530835, \text{ and } M_4N_4 = 0.006149663515. \end{cases} \tag{4.11}$$

With the use of Theorem 4.4, Table 3 presents the subdivision depth computations for various levels of iteration ℓ.

Table 3: Subdivision depth of 4-point approximating NSQSS.

ϵ	$5.42e^{-7}$					$3.07e^{-8}$				
$M_{\gamma_0^s}N_{\gamma_0^s}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
M_1N_1	73	73	73	73	73	89	88	88	88	88
M_2N_2	11	11	11	11	11	14	14	14	14	14
M_3N_3	8	8	8	8	8	10	9	9	9	9
M_4N_4	5	5	5	5	5	6	6	6	6	6
ϵ	$1.73e^{-9}$					$9.81e^{-11}$				
$M_{\gamma_0^s}N_{\gamma_0^s}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
M_1N_1	104	104	104	104	104	120	120	120	120	120
M_2N_2	16	16	16	16	16	19	19	19	19	19
M_3N_3	11	11	11	11	11	13	13	13	13	13
M_4N_4	7	7	7	7	7	9	9	9	9	9

Remark 4.6. In Table 3, the computation of subdivision depth at different levels ℓ corresponding to the pre-defined error tolerance is presented. For example, seventy-three iterations are required to attain a given error tolerance of $5.42e^{-7}$ by the method described in [16], but with the proposed method, only four iterations corresponding to M_4N_4 are needed. The graphical comparison of different convolutions is demonstrated in Figure 2 (a).

Example 4.7. Consider the tensor product of the 4-point interpolating NSQSS given in (3.3). Now apply four-times convolution to find the four convolution coefficients $M_{\gamma_0^s}N_{\gamma_0^s}$ using (4.5) and (4.6). These values are shown in (4.12).

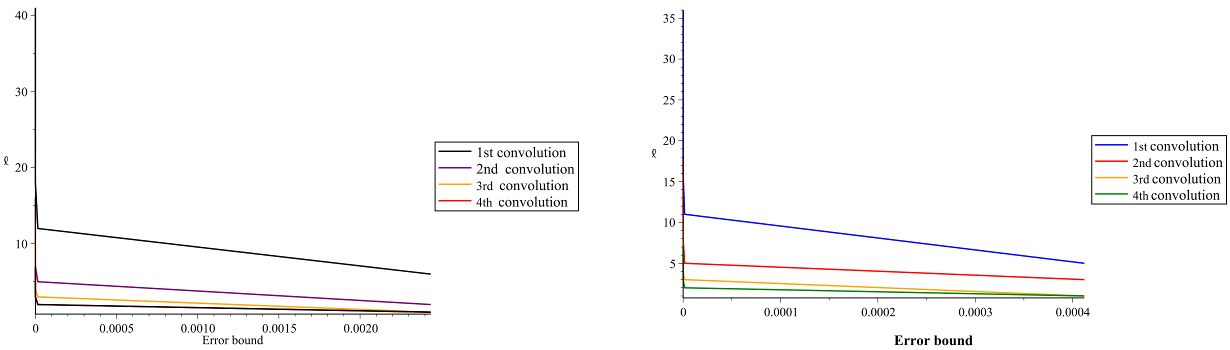
$$\begin{cases} \bullet \text{ Convolution coefficients for } \ell = 1,\ M_1N_1 = 0.3927105545,\ M_2N_2 = 0.09703817219, \\ \quad M_3N_3 = 0.02421449366, \text{ and } M_4N_4 = 0.006038140339. \\ \bullet \text{ Convolution coefficients for } \ell = 2,\ M_1N_1 = 0.3906970120,\ M_2N_2 = 0.09652801543, \\ \quad M_3N_3 = 0.02408796809, \text{ and } M_4N_4 = 0.006006608711. \\ \bullet \text{ Convolution coefficients for } \ell = 3,\ M_1N_1 = 0.3905722769,\ M_2N_2 = 0.09649628723, \\ \quad M_3N_3 = 0.02408005314, \text{ and } M_4N_4 = 0.006004638367. \\ \bullet \text{ Convolution coefficients for } \ell = 4,\ M_1N_1 = 0.3905644848,\ M_2N_2 = 0.09649430516, \\ \quad M_3N_3 = 0.02407955843, \text{ and } M_4N_4 = 0.006004515268. \\ \bullet \text{ Convolution coefficients for } \ell = 5,\ M_1N_1 = 0.3905639975,\ M_2N_2 = 0.09649418065, \\ \quad M_3N_3 = 0.02407952737, \text{ and } M_4N_4 = 0.006004507494. \end{cases} \tag{4.12}$$

The number of iterations for various levels ℓ is presented in Table 4. These values are computed using Theorem 4.4.

Table 4: Subdivision depth of 4-point interpolating NSQSS.

ϵ	$6.61e^{-2}$					$1.54e^{-3}$				
$M_{\gamma_0^s}N_{\gamma_0^s}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
M_1N_1	7	7	7	7	7	16	16	16	16	16
M_2N_2	2	2	2	2	2	7	7	7	7	7
M_3N_3	1	1	1	1	1	4	4	4	4	4
M_4N_4	1	1	1	1	1	3	3	3	3	3
ϵ	$3.60e^{-5}$					$8.42e^{-7}$				
$M_{\gamma_0^s}N_{\gamma_0^s}$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$	$\ell=5$
M_1N_1	26	25	25	25	25	35	35	35	35	35
M_2N_2	11	11	11	11	11	15	15	15	15	15
M_3N_3	7	7	7	7	7	9	9	9	9	9
M_4N_4	5	5	5	5	5	7	7	7	7	7

Remark 4.8. In Table 4, thirty-five iterations are necessary to achieve a given error tolerance of 8.42×10^{-7} using the technique given in [16]. However, with the proposed method, it requires only seven iterations corresponding to P_4Q_4. The graphical representation of different convolutions can be seen in Figure 2 (b).



(a) 4-point approximating NSQSS [1]. (b) 4-point interpolating NSQSS [28].

Figure 2: Figures (a) and (b) represents the comparison between different convolution results for the curve case. This shows that the error decreases with the increase of order of convolution. Here horizontal axis shows the error bound and vertical axis shows the subdivision level ℓ.

5. Algorithms to compute subdivision depths

In this subsection, we will provide algorithms for calculating the subdivision depths of schemes with varying complexity. These algorithms are critical for understanding the entire research work presented in this publication. Moreover, by applying these algorithms, readers will gain a thorough understanding of how to determine the subdivision depths, even when working with non-stationary schemes. In this paper, we provide five algorithms. The convolution coefficient and convolution computation are handled by Algorithms 1 and 2, respectively. The error bounds for curve and surface cases are determined by Algorithms 3 and 4, respectively. To estimate the required number of iterations with a specified error tolerance, Algorithm 5 is used for the curve case and Algorithm 6 for the surface case.

Algorithm 1 *The computation of convolution coefficient* $T_{\gamma_0^c}$.

Input: Arity of the subdivision scheme (4 for quaternary case), complexity of the subdivision scheme P, value of initial difference between two consecutive points, and subdivision level ℓ.

1. Define mask of the subdivision scheme
2. for $\hat{s}$ from 0 to $P-1$
3. calculate $\hat{h}^{\ell}_{4\hat{s}}$, $\hat{h}^{\ell}_{4\hat{s}+1}$, $\hat{h}^{\ell}_{4\hat{s}+2}$ and $\hat{h}^{\ell}_{4\hat{s}+3}$ that are defined in (2.4).
4. for $n < 4P$
5. calculate $\hat{h}^{\ell}_{n} = \{\hat{h}^{\ell}_{n}\}_{n=0}^{4P-1}$
6. else
7. if $n \geqslant 4P$ then
8. return zero
9. end if
10. end for
11. end for
12. for $\gamma_0^c \geqslant 1$, calculate $\Sigma(\gamma_0^c, P) = \{\Omega(\gamma_0^c, P) - 4^{\gamma_0^c} + 1, \Omega(\gamma_0^c, P) - 4^{\gamma_0^c} + 2, \dots, \Omega(\gamma_0^c, P)\}$, where $\Omega(\gamma_0^c, P) = (4^{\gamma_0^c} - 3)(4P-1)$.
13. end for
14. Now compute $T_{\gamma_0^c} = \max_{j\in\Sigma(\gamma_0^c,P)}\left\{\sum_{\hat{s}=0}^{\lfloor j/4^{\gamma_0^c}\rfloor} |S^{\gamma_0^c;\hat{h}^{\ell}}_{\hat{s},j}|\right\}$
15. For finite sequence of ℓ and γ_0^c repeat steps 1 to 14.

Output: The values of convolution coefficient $T_{\gamma_0^c}$ at different values of ℓ.

6. Conclusion

This research focuses on the latest approach to computing error bounds and subdivision depth for non-stationary quaternary subdivision schemes using a convolution methodology. This method yields excellent results with minimal computational cost and does not depend on specific conditions regarding the coefficients of non-stationary quaternary subdivision schemes. Initially, associations between constants and the vectors generated by these non-stationary schemes were established, followed by the formulation of an expression for the convolution. This expression demonstrates values that consistently decrease as the convolution order increases, applicable to both curves and surfaces. Our method is robust and performs well with all types of data, creating regular initial control polygons. Increasing the order of convolution enhances the robustness of the error bounds.

Algorithm 2 *The computation of convolution* $M_{\gamma_0^s} N_{\gamma_0^s}$.

Input: Arity of the subdivision scheme (4 in quaternary case), complexity of the subdivision scheme P, value of initial difference between two consecutive points, and subdivision level ℓ.

1. Define mask of the subdivision scheme
2. for $\hat{r}$ from 0 to $P-1$ do
3. calculate $E^{\ell}_{4\hat{r}}$, $E^{\ell}_{4\hat{r}+1}$, $E^{\ell}_{4\hat{r}+2}$ and $E^{\ell}_{4\hat{r}+3}$ that are defined in (4.2)
4. for $\hat{a}$ from 0 to $P-1$ do
5. calculate $F^{\ell}_{4\hat{a}}$, $F^{\ell}_{4\hat{a}+1}$, $F^{\ell}_{4\hat{a}+2}$ and $F^{\ell}_{4\hat{a}+3}$ that are defined in (4.2)
6. for $n < 4P$ do
7. calculate $E^{\ell}_{n} = \{E^{\ell}_{n}\}_{n=0}^{4P-1}$ and $F^{\ell} = \{F^{\ell}_{n}\}_{n=0}^{4P-1}$
8. else
9. if $n \geqslant 4P$ then
10. return zero
11. end if
12. end for
13. end for
14. end for
15. for $\gamma_0^s \geqslant 1$ calculate $\Sigma(\gamma_0^s, P) = \{\Omega(\gamma_0^s, P) - 4^{\gamma_0^s} + 1, \Omega(\gamma_0^s, P) - 4^{\gamma_0^s} + 2, \ldots, \Omega(\gamma_0^s, P)\}$, where $\Omega(\gamma_0^s, P) = (4^{\gamma_0^s} - 3)(4P - 1)$.
16. end for
17. compute $M_{\gamma_0^c} N_{\gamma_0^c}$, where
$$M_{\gamma_0^s} = \max_i \left\{ \sum_{m=0}^{\lfloor i/4^{\gamma_0^s} \rfloor} |S_{m,i}^{\gamma_0^s, E^{\ell}}| \right\} \quad \text{and} \quad N_{\gamma_0^s} = \max_j \left\{ \sum_{n=0}^{\lceil j/4^{\gamma_0^s} \rceil} |S_{n,j}^{\gamma_0^s, F^{\ell}}| \right\}.$$
18. For finite sequence of ℓ and γ_0^s repeat steps 1 to 17.

Output: The values of convolution $M_{\gamma_0^c} N_{\gamma_0^c}$ at different values of ℓ.

Algorithm 3 *The error bounds and level of subdivision* ℓ *for curve case.*

Input: Arity, complexity, subdivision Level $\ell \geqslant 1$, order of convolution $\gamma_0^c \geqslant 1$, and the value of convolution coefficient $T_{\gamma_0^c}$.

1. Mask of the schemes
2. for $\gamma_0^c \in \mathbb{N}$ do
3. for $\ell \in \mathbb{N}$ do
4. compute $\varpi \left(\frac{(T_{\gamma_0^c})^{\ell}}{1 - T_{\gamma_0^c}} \right) \kappa$.
5. end do (ℓ)
6. end do (γ_0^c)

Output: The error bounds at $\ell \geqslant 1$ and $\gamma_0^c \geqslant 1$.

Algorithm 4 *The error bounds and level of subdivision ℓ for surface case.*

Input: Arity, complexity, subdivision Level $\ell \geqslant 1$, order of convolution $\gamma_0^s \geqslant 1$, and the value of convolution coefficient $M_{\gamma_0^s} N_{\gamma_0^s}$.

1. for $\gamma_0^s \in \mathbb{N}$ do
2. for $\ell \in \mathbb{N}$ do
3. compute $\vartheta \left(\frac{(M_{\gamma_0^s} N_{\gamma_0^s})^\ell}{1 - M_{\gamma_0^s} N_{\gamma_0^s}} \right)$.
4. end do (ℓ)
5. end do (γ_0^s)

Output: The error bounds at $\ell \geqslant 1$ and $\gamma_0^s \geqslant 1$.

Algorithm 5 *The subdivision depths at pre-defined error tolerance at different level of subdivision ℓ for curve case.*

Input: $\gamma_0^c \geqslant 1$, $T_{\gamma_0^c}$, κ, and ϵ.

1. for $\gamma_0^c \in \mathbb{N}$ do
2. compute $\log_{(T_{\gamma_0^c})^{-1}} \left(\frac{\varpi \kappa}{\epsilon (1 - T_{\gamma_0^c})} \right) \leftarrow \ell$
3. end do

Output: The subdivision depths ℓ at pre-defined error tolerance ϵ, and $\gamma_0^c \geqslant 1$.

Algorithm 6 *The subdivision depths at pre-defined error tolerance for different level of subdivision ℓ for surface case.*

Input: $\gamma_0^s \geqslant 1$, $P_{\gamma_0^s} Q_{\gamma_0^s}$, ν and ϵ.

1. for $\gamma_0^s \in \mathbb{N}$ do
2. compute $\log_{(M_{\gamma_0^s} N_{\gamma_0^s})^{-1}} \left(\frac{\vartheta}{\epsilon (1 - M_{\gamma_0^s} N_{\gamma_0^s})} \right) \leftarrow \ell$
3. end do

Output: The subdivision depths ℓ at pre-defined error tolerance ϵ, and $\gamma_0^s \geqslant 1$.

References

[1] P. Ashraf, G. Mustafa, *A generalized non-stationary 4-point b-ary approximating scheme*, Br. J. Math. Comput. Sci., **4** (2014), 104–119. 1, 1.1, 2.2, 3.4, 1a, 2a

[2] P. Ashraf, G. Mustafa, A. Ghaffar, R. Zahra, K. S. Nisar, E. E. Mahmoud, W. R. Alharbi, *Unified framework of approximating and interpolatory subdivision schemes for construction of class of binary subdivision schemes*, J. Funct. Spaces, **2020** (2020), 12 pages. 1

[3] P. Ashraf, G. Mustafa, H. A. Khan, D. Baleanu, A. Ghaffar, K. S. Nisar, *A shape-preserving variant of Lane-Riesenfeld algorithm*, AIMS Math., **6** (2021), 2152–2170.

[5] M. Bari, G. Mustafa, A. Ghaffar, K. S. Nisar, D. Baleanu, *Construction and analysis of unified 4-point interpolating nonstationary subdivision surfaces*, Adv. Differ. Equ., **2021** (2021), 17 pages.

[4] C. Beccari, G. Casciola, L, Romani, *A non-stationary uniform tension controlled interpolating 4-point scheme reproducing conics*, Comput. Aided Geom. Des., **24** (2007), 1–9.

[8] Z. J. Cai, *Convergence, error estimation and some properties of four-point interpolation subdivision scheme*, Comput. Aided Geom. Des., **12** (1995), 459–468. 1

[6] A. S. Cavaretta, W. Dahmen, C. A. Micchelli, *Stationary subdivision*, Mem. Amer. Math. Soc., (1991). 1

[9] S. W. Choi, B.-G. Lee, Y. J. Lee, J. Yoon, *Stationary subdivision schemes reproducing polynomials*, Comput. Aided Geom. Des., **23** (2006) 351-360. 1

[7] F. Cirak, M. J. Scott, E. K. Antonsson, M. Ortiz, P. Schroder, *Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision*, Comput. Aided Des., **34** (2002), 137–148. 1

[10] N. Dyn, *Subdivision schemes in CAGD*, Adv. Numer. Anal., **2** (1992), 36–104. 1

[12] N. Dyn, *Linear and non-linear subdivision schemes in geometric modeling*, Found. Comput. Math., Hong Kong, **363** (2008), 68–92. 1

[13] N. Dyn, J. A. Gregory, D. Levin, *Analysis of uniform binary subdivision schemes for curve design*, Constr. Approx., **7** (1991), 127–147. 1

[11] N. Dyn, D. Levin, *Subdivision schemes in geometric modeling*, Acta Numer., **11** (2002), 73–144. 1

[14] M.-e. Fang, B. Jeong, J. Yoon, *A family of non-uniform subdivision schemes with variable parameters for curve design*, Appl. Math. Comput., **313** (2017), 1–11. 1

[16] S. Hashmi, G. Mustafa, *Estimating error bounds for quaternary subdivision schemes*, J. Math. Anal. Appl., **358** (2009), 159–167. 2.2, 2.2, 3, 3.5, 3.6, 3.8, 4.1, 4.2, 4.6, 4.8

[15] W. Huawei, G. Youjiang, Q. Kaihuai, *Error estimation for Doo-Sabin surfaces*, Prog. Nat. Sci., **12** (2002), 697–700. 1

[17] S. A. A. Karim, F. Khan, G. Mustafa, A. Shahzad, M. Asghar, *An efficient computational approach for computing subdivision depth of non-stationary binary subdivision schemes*, Mathematics, **11** (2023), 1–12. 1

[21] M. Moncayo, S. Amat, *Error bounds for a class of subdivision schemes based on the two-scale refinement equation*, J. Comput. Appl. Math., **236** (2011), 265–278. 1

[19] M. Moncayo, J. F. Reinoso, S. Amat, *Tight numerical bounds for digital terrain modeling by interpolatory subdivision schemes*, Math. Comput. Simul., **81** (2011), 2258–2269. 1

[18] P. Morin, K. G. Siebert, A. Veeser, *A basic convergence result for conforming adaptive finite elements*, Math. Models Methods Appl. Sci., **18** (2008), 707–737. 1

[20] G. Mustafa, F. Chen, J. Deng, *Estimating error bounds for binary subdivision curves/surfaces*, J. Comput. Appl. Math., **193** (2006), 596–613. 1, 2.2

[22] G. Mustafa, M. S. Hashmi, *Subdivision depth computation for* n*-ary subdivision curves/surfaces*, Vis. Comput., **26** (2010), 841–851.1

[23] G. Mustafa, M. S. Hashmi, *Subdivision depth computation for tensor product* n*-ary volumetric models*, Abstr. Appl. Anal., **2011** (2011), 22 pages. 1

[24] G. Mustafa, S. Hashmi, N. A. Noshi, *Estimating error bounds for tensor product binary subdivision volumetric model*, Int. J. Comput. Math., **83** (2006), 879–903.1

[25] A. Nawaz, A. Ghaffar, F. Khan, S. A. A. Karim, *A new 7-point quaternary approximating subdivision scheme*, In: Intelli. Sys. Model. Simul. II., Springer, Cham, **444** (2022), 545–566. 1

[26] M. Perini, P. Bosetti, N. Balc, *Additive manufacturing for repairing: from damage identification and modeling to Direct Laser Deposition*, Rapid Prototyp. J., **26** (2020), 929–940. 1

[27] A. Shahzad, F. Khan, A. Ghaffar, S. W. Yao, M. Inc, S. Ali, *A novel numerical method for computing subdivision depth of quaternary schemes*, Mathematics, **9** (2021), 1–20. 1, 1.1

[28] S. S. Siddiqi, M. Younis, *The quaternary interpolating scheme for geometric design*, ISRN Comput. Graph., **2013** (2013), 8 pages. 1, 1.1, 2.2, 3.7, 1b, 2b

[29] N. Umetani, B. Bickel, *Learning three-dimensional flow for interactive aerodynamic design*, ACM Trans. Graph. (TOG)., **37** (2018), 1–10. 1

[30] L. Wang, *Integration of CAD and boundary element analysis through subdivision methods*, Comput. Ind. Eng., **57** (2009), 691–698. 1

[31] G. Zhou, X.-M. Zeng, *Error bounds for Loop subdivision surfaces*, Int. J. Comput. Math., **91** (2014), 688–703. 1