IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 April 2024, accepted 24 May 2024, date of publication 31 May 2024, date of current version 10 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3407849

RESEARCH ARTICLE

Establishing a Specialized Bridge Between the Even-Point Binary and the Even/Odd-Point Quaternary Subdivision Approaches

SIDRA NOSHEEN[1], RABIA HAMEED[1], AND JIHAD YOUNIS[2]
[1]Department of Mathematics, The Government Sadiq College Women University Bahawalpur, Bahawalpur 63100, Pakistan
[2]Department of Mathematics, Aden University, Aden, Yemen

Corresponding authors: Jihad Younis (jihadalsaqqaf@gmail.com) and Rabia Hameed (rabia.hameed@gscwu.edu.pk)

ABSTRACT Subdivision schemes are powerful tools for generating curves and surfaces in computer graphics. This work explores a novel connection between binary and quaternary schemes, where quaternary schemes can be derived from binary ones. We present a generalized formula for constructing $(3n-1)$-point quaternary schemes from existing $2n$-point binary schemes. This approach leads to two types of quaternary schemes based on even and odd values of n. We demonstrate the efficiency of these new schemes by applying them to known binary schemes and analyzing their properties. Our results show that the derived quaternary schemes achieve similar final models as their binary counterparts, but with fewer iterations, leading to significant computational cost reduction. This effectiveness is validated through graphical and theoretical analyses, confirming the applicability of our method to both parametric and non-parametric settings.

INDEX TERMS Binary subdivision scheme, quaternary subdivision scheme, Hölder's regularity, degree of precision, mask.

I. INTRODUCTION

Subdivision methods for curves were introduced and mathematically analyzed for the first time by de Rham [9] in 1956 and re-invent for computer graphics community by Chaikin [8] in 1974. Subdivision is actually an iterative method to generate smooth curves and surfaces. Subdivision schemes increase the points at each iteration to get smooth shapes. If the subdivision process increases points two times at each iteration then this process is known as the binary subdivision process, whereas if a subdivision process increases points four times at each iteration then this process is known as quaternary subdivision process. The tools which are used for the binary and quaternary subdivision processes are known as the binary and quaternary subdivision schemes respectively.

Mathematically, the general compact forms of univariate r-ary subdivision scheme which is used to get a refined polygon $G^{k+1} = \{g_i^{k+1}\}_{i\in\mathbb{Z}} \in n(\mathbb{Z})$ from the polygon $g^k = \{g_i^k\}_{i\in\mathbb{Z}} \in n(\mathbb{Z})$ can be defined in terms of a mask consisting of a finite set of non-zero coefficients $\beta = \{\beta_j\}_{j\in\mathbb{Z}}$ as follows:

$$g^{k+1}_{r\phi+\eta} = \sum_{j\in\mathbb{Z}} \beta_{rj+\eta}\ g^k_{\phi+j}, \tag{1}$$

where the set of values $\{r = 2, \ \eta = -1, 0\}$ are for the binary subdivision rules and the set of values $\{r = 4, \eta = -2, -1, 0, 1\}$ are used for the quaternary subdivision rules respectively, $n(\mathbb{Z})$ denote the space of scaler-valued sequences. The sequence $\beta = \{\beta_j\}_{j\in\mathbb{Z}}$ is called the refinement mask. The polynomial which uses this mask as coefficients is called the Laurent's polynomial. Therefore, the Laurent polynomial corresponding to subdivision scheme (1) is

$$\mu(c) = \sum_{j\in\mathbb{Z}} \beta_{rj+\eta}\ c^{rj+\eta}.$$

A convergent subdivision scheme with the corresponding mask $\beta = \{\beta_j\}_{j\in\mathbb{Z}}$ necessarily satisfies the following convergence condition:

$$\sum_{j\in\mathbb{Z}} \beta_{rj+\eta} = 1.$$

The associate editor coordinating the review of this manuscript and approving it for publication was Luca Cassano.

Binary subdivision schemes were first introduced by Dyn et al. [20], who also analyzed the convergence of control polygons produced by these schemes. Since then, significant advancements have been made. Ghaffar et al. introduced a novel class of non-stationary binary schemes using Lagrange interpolation, offering advantages in specific applications compared to traditional stationary schemes [3]. Hameed and Mustafa proposed a variant of the Lane-Riesenfeld algorithm with a modified refining operator, expanding the options for curve and surface generation [24]. Mustafa et al. explored free-parameter binary approximating schemes, demonstrating that odd-point schemes often achieve better error bounds [11]. Mustafa and Hameed further developed univariate and bivariate schemes using a two-step algorithm that combines quartic B-spline refinement with point averaging. This approach allows for the creation of primal and dual schemes based on the number of smoothing steps [13]. Finally, Siddiqi and Younis presented a binary univariate scheme that generates limit curves and established a method for deriving B-spline blending functions, providing valuable tools for further research [30]. The analysis of binary subdivision schemes has been addressed by several researchers, including [1], [10], [19], [22], [23], [24], [32].

Several advancements have been made in quaternary subdivision schemes. Bari et al. introduced a method for generating efficient $3n$-point schemes and explored properties like polynomial reproduction and the ability to preserve shapes (monotonicity, convexity, concavity) [17]. Hashmi and Mustafa developed procedures for estimating errors independent of the subdivision process itself, offering valuable tools for analysis [27]. Mustafa et al. proposed a versatile family of quaternary schemes with multiple parameters, along with their convergence conditions [3]. Mustafa et al. focused on a specific 7-point scheme with a shape parameter, analyzing its continuity, smoothness (Hölder's regularity), and limiting stencils [12]. Siddiqi and Younis presented a construction method using the Cox-de Boor formula, expressing the schemes in terms of B-spline blending functions, a widely used approach [31]. Shahzad et al. explored applications of quaternary schemes in calculating subdivision depth and error bounds for complex surface models [2]. Despite these advancements, a deeper understanding of the relationship between binary and quaternary schemes remains a gap in the current literature. Our work aims to address this by investigating these connections. By bridging this gap, we hope to contribute new insights and potentially lead to novel approaches in subdivision scheme development.

A. CONTRIBUTION AND NOVELTY OF THE PROPOSED APPROACH

The contribution and novelty of our research are thoroughly covered in this section, along with the shortcomings of the state-of-the-art methods and how our suggested solution fills in their gaps.

1) PROBLEM STATEMENT AND STATE-OF-THE-ART METHODS

Subdivision schemes are essential in computer graphics and geometric modeling, but traditional binary and quaternary schemes have limitations.

Binary schemes require more iterations, increasing computational costs and processing times. Current methods focus on optimizing binary schemes to reduce iterations, but these often involve complex calculations and may not be universally applicable to all geometric data types.

Quaternary subdivision schemes can achieve desired smoothness in fewer iterations, but designing them from scratch is complex and there's no systematic approach to derive them from existing binary schemes. Current research has made progress, but the connection between binary and quaternary schemes remains under-explored.

2) GAP IN EXISTING RESEARCH

The main deficiency in the existing literature is the lack of a systematic and comprehensive approach for deriving quaternary subdivision schemes from binary ones. This gap prevents the use of proven binary schemes as a basis for quaternary scheme development.

3) CONTRIBUTION OF OUR WORK

The research focuses on creating a bridge between even-point binary and even/odd-point quaternary subdivision approaches. It reveals a novel relation between the two, simplifying the process of creating quaternary schemes from existing binary schemes. The authors present a generalized formula for subdivision rules for $(3n - 1)$-point quaternary approximating subdivision schemes, making the design process more efficient and systematic. The results are validated by applying the formula to known binary schemes for specific values of n, demonstrating that derived quaternary schemes achieve desired detail and smoothness in fewer iterations, reducing computational costs and processing times in practical applications.

4) NOVELTY AND MOTIVATION

Our work is novel because of the creative way we use a generalised relation to connect binary and quaternary subdivision schemes. By utilising the advantages of both kinds of schemes, this method presents a methodical way to derive quaternary schemes from binary ones.

Our desire to increase the efficacy and efficiency of subdivision schemes in geometric modelling is what drives us. Our approach to methodically derive quaternary schemes from binary ones that already exist makes us a useful tool for researchers and practitioners, allowing them to expand on existing knowledge and accomplish better results with less work.

Finally, we conclude that our work significantly advances the state-of-the-art in geometric modelling by providing a novel, effective, and workable solution that bridges the gap between binary and quaternary subdivision schemes.

More precisely, the paper is organised as follows: In Section II, we give the generalized procedure to define the relation between the binary and the quaternary subdivision schemes. In Section III, we give the applications of the given technique along with the graphical comparisons of the pair of the binary and quaternary subdivision schemes. Section IV is about the Hölder's regularity computation of the binary and the quaternary subdivision schemes. In Section V, we give the response of these pairs of subdivision scheme on the polynomial data. Conclusions and future trends are discussed in Section VI.

II. LINK BETWEEN THE BINARY AND QUATERNARY SUBDIVISION SCHEMES

This section explains a new observation about the relation between the even-point binary approximating subdivision schemes and the quaternary approximating subdivision schemes. The generalized subdivision rules of $(3n-1)$-point quaternary subdivision schemes is deduced by using the subdivision rules of the $2n$-point binary subdivision schemes. The $2n$-point dual binary subdivision scheme which maps the polygon g to a refined polygon G after one level of refinement can be written as:

$$\begin{cases} g^{k+1}_{2\varphi-1} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2\lambda}\ g^{k}_{\varphi+\lambda-1}, \\ g^{k+1}_{2\varphi} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2-2\lambda}\ g^{k}_{\varphi+\lambda}, \end{cases} \tag{2}$$

where $\{g^k_{\varphi+\lambda} : \lambda = -n+1, \ldots, n : \varphi \in \mathbb{R}\}$ are the control points at k-th subdivision level and $\{\beta_\lambda : \lambda = -2n+2, \ldots, 2n\}$ is the mask of the subdivision scheme.

The following lemma gives a new form of the subdivision rules of $2n$-point binary subdivision scheme which is defined in (2).

Lemma 1: If we change φ by the odd numbers $2\varphi-1$ and $2\varphi+1$ and the even number 2φ in the subdivision equations of the binary subdivision scheme (2), then these subdivision equations reduced into the four subdivision equations.

Proof: Firstly, we re-write the subdivision scheme (2) in the form which is free from subdivision levels, hence we get two subdivision equations given below:

$$\begin{cases} g_{2\varphi-1} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2\lambda}\ g_{\varphi+\lambda-1}, \\ g_{2\varphi} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2-2\lambda}\ g_{\varphi+\lambda}, \end{cases} \tag{3}$$

where $\varphi \in \mathbb{R}$.

We now perform substitutions in the second subdivision equation of (3). We replace all instances of φ by $2\varphi-1$. Additionally, in both subdivision equations of (3), we substitute φ by 2φ. Finally, in the first subdivision equation of (3), we replace φ by $2\varphi+1$. This substitution process leads to the following four equations:

$$\begin{cases} g_{4\varphi-2} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2-2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi-1} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2-2\lambda}\ g_{2\varphi+\lambda}, \\ g_{4\varphi+1} = \sum\limits_{\lambda=-n+1}^{n} \beta_{2\lambda}\ g_{2\varphi+\lambda}. \end{cases} \tag{4}$$

Which completes the proof. □

Now we split the further process into two parts depending on the even and odd values of n. The first theorem is proved for the even values of n, while the second one is proved for the odd n.

Theorem 2: If n is even, that is $n = 2m : m \in \mathbb{N}$, then the subdivision rules $g_{4\varphi-2}$ and $g_{4\varphi-1}$ in (4) are the linear combination of $6m-1$ control points $g_{\varphi-3m+1} \cdots g_{\varphi+3m-1}$, while the subdivision rules $g_{4\varphi}$ and $g_{4\varphi+1}$ in (4) are the linear combination of $6m$ control points $g_{\varphi-3m+1} \cdots g_{\varphi+3m}$.

Proof: Since n is even, so firstly we put $n = 2m$ in (4). Thus we get

$$\begin{cases} g_{4\varphi-2} = \sum\limits_{\lambda=-2m+1}^{2m} \beta_{2-2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi-1} = \sum\limits_{\lambda=-2m+1}^{2m} \beta_{2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi} = \sum\limits_{\lambda=-2m+1}^{2m} \beta_{2-2\lambda}\ g_{2\varphi+\lambda}, \\ g_{4\varphi+1} = \sum\limits_{\lambda=-2m+1}^{2m} \beta_{2\lambda}\ g_{2\varphi+\lambda}. \end{cases} \tag{5}$$

Now we have to find out values of $g_{2\varphi-2m}, g_{2\varphi-2m+1}, \ldots, g_{2\varphi+2m-1}, g_{2\varphi+2m}$. For this, first we evaluate (3) for $n = 2m$ and then by increasing or decreasing the subscript, we get the required unknowns.

$$\begin{cases} g_{2\varphi-1} = \beta_{2-4m}\ g_{\varphi-2m} + \beta_{4-4m}\ g_{\varphi-2m+1} \\ \quad +\beta_{6-4m}\ g_{\varphi-2m+2} + \beta_{8-4m}\ g_{\varphi-2m+3} \\ \quad +\beta_{10-4m}\ g_{\varphi-2m+4} + \beta_{12-4m}\ g_{\varphi-2m+5} \\ \quad +\ldots+\beta_{4m-8}\ g_{\varphi+2m-5} + \beta_{4m-6} \\ \quad \times g_{\varphi+2m-4} + \beta_{4m-4}\ g_{\varphi+2m-3} + \beta_{4m-2} \\ \quad \times g_{\varphi+2m-2} + \beta_{4m}\ g_{\varphi+2m-1}, \\ g_{2\varphi} = \beta_{4m}\ g_{\varphi-2m+1} + \beta_{4m-2}\ g_{\varphi-2m+2} \\ \quad +\beta_{4m-4}\ g_{\varphi-2m+3} + \beta_{4m-6}\ g_{\varphi-2m+4} \\ \quad +\beta_{4m-8}\ g_{\varphi-2m+5} + \beta_{4m-10}\ g_{\varphi-2m+6} \\ \quad +\ldots+\beta_{10-4m}\ g_{\varphi+2m-4} + \beta_{8-4m} \\ \quad \times g_{\varphi+2m-3} + \beta_{6-4m}\ g_{\varphi+2m-2} + \beta_{4-4m} \\ \quad g_{\varphi+2m-1} + \beta_{2-4m}\ g_{\varphi+2m}. \end{cases} \tag{6}$$

Now we replace φ by $\varphi - m$ in second rule of (6), we get

$$\begin{aligned} g_{2\varphi-2m} &= \beta_{4m}\ g_{\varphi-3m+1} + \beta_{4m-2}\ g_{\varphi-3m+2} + \beta_{4m-4} \\ &\times g_{\varphi-3m+3} + \beta_{4m-6}\ g_{\varphi-3m+4} + \beta_{4m-8} \\ &\times g_{\varphi-3m+5} + \beta_{4m-10} g_{\varphi-3m+6} + \ldots + \beta_{10-4m} \\ &\times g_{\varphi+m-4} + \beta_{8-4m} g_{\varphi+m-3} + \beta_{6-4m} g_{\varphi+m-2} \\ &+ \beta_{4-4m}\ g_{\varphi+m-1} + \beta_{2-4m}\ g_{\varphi+m}. \end{aligned}$$

Now we replace φ by $\varphi - m + 1$ in the first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi-2m+1} &= \beta_{2-4m} g_{\varphi-3m+1} + \beta_{4-4m} g_{\varphi-3m+2} + \beta_{6-4m} \\ &\times g_{\varphi-3m+3} + \beta_{8-4m} g_{\varphi-3m+4} + \beta_{10-4m} g_{\varphi-3m+5} \\ &+ \beta_{12-4m}\ g_{\varphi-3m+6} + \ldots + \beta_{4m-8}\ g_{\varphi+m-4} \\ &+ \beta_{4m-6}\ g_{\varphi+m-3} + \beta_{4m-4}\ g_{\varphi+m-2} + \beta_{4m-2} \\ &\times g_{\varphi+m-1} + \beta_{4m}\ g_{\varphi+m}, \\ g_{2\varphi-2m+2} &= \beta_{4m}\ g_{\varphi-3m+2} + \beta_{4m-2}\ g_{\varphi-3m+3} + \beta_{4m-4} \\ &\times g_{\varphi-3m+4} + \beta_{4m-6} g_{\varphi-3m+5} + \beta_{4m-8} g_{\varphi-3m+6} \\ &+ \beta_{4m-10}\ g_{\varphi-3m+7} + \ldots + \beta_{10-4m}\ g_{\varphi+m-3} \\ &+ \beta_{8-4m}\ g_{\varphi+m-2} + \beta_{6-4m}\ g_{\varphi+m-1} + \beta_{4-4m} \\ &\times g_{\varphi+m} + \beta_{2-4m}\ g_{\varphi+m+1}. \end{aligned}$$

Now we replace φ by $\varphi - m + 2$ in the first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi-2m+3} &= \beta_{2-4m}\ g_{\varphi-3m+2} + \beta_{4-4m} g_{\varphi-3m+3} + \beta_{6-4m} \\ &\times g_{\varphi-3m+4} + \beta_{8-4m} g_{\varphi-3m+5} + \beta_{10-4m} g_{\varphi-3m+6} \\ &+ \beta_{12-4m}\ g_{\varphi-3m+7} + \ldots + \beta_{4m-8}\ g_{\varphi+m-3} \\ &+ \beta_{4m-6}\ g_{\varphi+m-2} + \beta_{4m-4}\ g_{\varphi+m-1} + \beta_{4m-2} \\ &\times g_{\varphi+m} + \beta_{4m}\ g_{\varphi+m+1}, \\ g_{2\varphi-2m+4} &= \beta_{4m}\ g_{\varphi-3m+3} + \beta_{4m-2}\ g_{\varphi-3m+4} + \beta_{4m-4} \\ &\times g_{\varphi-3m+5} + \beta_{4m-6}\ g_{\varphi-3m+6} + \beta_{4m-8} g_{\varphi-3m+7} \\ &+ \beta_{4m-10}\ g_{\varphi-3m+8} + \ldots + \beta_{10-4m}\ g_{\varphi+m-2} \\ &+ \beta_{8-4m} g_{\varphi+m-1} + \beta_{6-4m} g_{\varphi+m} + \beta_{4-4m} g_{\varphi+m+1} \\ &+ \beta_{2-4m}\ g_{\varphi+m+2}. \end{aligned}$$

Now we replace φ by $\varphi - m + 3$ in first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi-2m+5} &= \beta_{2-4m}\ g_{\varphi-3m+3} + \beta_{4-4m}\ g_{\varphi-3m+4} + \beta_{6-4m} \\ &\times g_{\varphi-3m+5} + \beta_{8-4m} g_{\varphi-3m+6} + \beta_{10-4m} g_{\varphi-3m+7} \\ &+ \beta_{12-4m}\ g_{\varphi-3m+8} + \ldots + \beta_{4m-8}\ g_{\varphi+m-2} \\ &+ \beta_{4m-6} g_{\varphi+m-1} + \beta_{4m-4} g_{\varphi+m} + \beta_{4m-2} g_{\varphi+m+1} \\ &+ \beta_{4m}\ g_{\varphi+m+2}, \\ g_{2\varphi-2m+6} &= \beta_{4m}\ g_{\varphi-3m+4} + \beta_{4m-2}\ g_{\varphi-3m+5} + \beta_{4m-4} \\ &\times g_{\varphi-3m+6} + \beta_{4m-6}\ g_{\varphi-3m+7} + \beta_{4m-8} g_{\varphi-3m+8} \\ &+ \beta_{4m-10}\ g_{\varphi-3m+9} + \ldots + \beta_{10-4m} g_{\varphi+m-1} \\ &+ \beta_{8-4m}\ g_{\varphi+m} + \beta_{6-4m}\ g_{\varphi+m+1} + \beta_{4-4m} \\ &\times g_{\varphi+m+2} + \beta_{2-4m}\ g_{\varphi+m+3}. \\ &\vdots \qquad\qquad \vdots \end{aligned}$$

Continuing this process, we replace φ by $\varphi + m - 2$ in first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi+2m-5} &= \beta_{2-4m}\ g_{\varphi-m-2} + \beta_{4-4m}\ g_{\varphi-m-1} + \beta_{6-4m} \\ &\times g_{\varphi-m} + \beta_{8-4m}\ g_{\varphi-m+1} + \beta_{10-4m}\ g_{\varphi-m+2} \\ &+ \beta_{12-4m}\ g_{\varphi-m+3} + \ldots + \beta_{4m-8}\ g_{\varphi+3m-7} \\ &+ \beta_{4m-6}\ g_{\varphi+3m-6} + \beta_{4m-4}\ g_{\varphi+3m-5} + \beta_{4m-2} \\ &\times g_{\varphi+3m-4} + \beta_{4m}\ g_{\varphi+3m-3}, \\ g_{2\varphi+2m-4} &= \beta_{4m}\ g_{\varphi-m-1} + \beta_{4m-2}\ g_{\varphi-m} + \beta_{4m-4}\ g_{\varphi-m+1} \\ &+ \beta_{4m-6}\ g_{\varphi-m+2} + \beta_{4m-8}\ g_{\varphi-m+3} + \beta_{4m-10} \\ &\times g_{\varphi-m+4} + \ldots + \beta_{10-4m}\ g_{\varphi+3m-6} + \beta_{8-4m} \\ &\times g_{\varphi+3m-5} + \beta_{6-4m} g_{\varphi+3m-4} + \beta_{4-4m} g_{\varphi+3m-3} \\ &+ \beta_{2-4m}\ g_{\varphi+3m-2}. \end{aligned}$$

Now we replace φ by $\varphi + m - 1$ in the first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi+2m-3} &= \beta_{2-4m}\ g_{\varphi-m-1} + \beta_{4-4m}\ g_{\varphi-m} + \beta_{6-4m} \\ &\times g_{\varphi-m+1} + \beta_{8-4m}\ g_{\varphi-m+2} + \beta_{10-4m} \\ &\times g_{\varphi-m+3} + \beta_{12-4m}\ g_{\varphi-m+4} + \ldots + \beta_{4m-8} \\ &\times g_{\varphi+3m-6} + \beta_{4m-6}\ g_{\varphi+3m-5} + \beta_{4m-4} \\ &\times g_{\varphi+3m-4} + \beta_{4m-2} g_{\varphi+3m-3} + \beta_{4m} g_{\varphi+3m-2}, \\ g_{2\varphi+2m-2} &= \beta_{4m}\ g_{\varphi-m} + \beta_{4m-2}\ g_{\varphi-m+1} + \beta_{4m-4} \\ &\times g_{\varphi-m+2} + \beta_{4m-6} g_{\varphi-m+3} + \beta_{4m-8} g_{\varphi-m+4} \\ &+ \beta_{4m-10} g_{\varphi-m+5} + \ldots + \beta_{12-4m} g_{\varphi+3m-6} \\ &+ \beta_{10-4m}\ g_{\varphi+3m-5} + \beta_{8-4m} g_{\varphi+3m-4} \\ &+ \beta_{6-4m} g_{\varphi+3m-3} + \beta_{4-4m} g_{\varphi+3m-2} + \beta_{2-4m} \\ &\times g_{\varphi+3m-1}. \end{aligned}$$

Now we replace φ by $\varphi + m$ in first and second rules of (6), we get

$$\begin{aligned} g_{2\varphi+2m-1} &= \beta_{2-4m}\ g_{\varphi-m} + \beta_{4-4m}\ g_{\varphi-m+1} + \beta_{6-4m} \\ &\times g_{\varphi-m+2} + \beta_{8-4m}\ g_{\varphi-m+3} + \beta_{10-4m} \\ &\times g_{\varphi-m+4} + \beta_{12-4m} g_{\varphi-m+5} + \ldots + \beta_{4m-10} \\ &\times g_{\varphi+3m-6} + \beta_{4m-8}\ g_{\varphi+3m-5} + \beta_{4m-6} \\ &\times g_{\varphi+3m-4} + \beta_{4m-4} g_{\varphi+3m-3} + \beta_{4m-2} g_{\varphi+3m-2} \\ &+ \beta_{4m}\ g_{\varphi+3m-1}, \end{aligned}$$

$$g_{2\varphi+2m} = \beta_{4m}\ g_{\varphi-m+1} + \beta_{4m-2}\ g_{\varphi-m+2} + \beta_{4m-4} \times g_{\varphi-m+3} + \beta_{4m-6}g_{\varphi-m+4} + \beta_{4m-8}g_{\varphi-m+5} + \beta_{4m-10}g_{\varphi-m+6} + \ldots + \beta_{14-4m}g_{\varphi+3m-6} + \beta_{12-4m}\ g_{\varphi+3m-5} + \beta_{10-4m}\ g_{\varphi+3m-4} + \beta_{8-4m}g_{\varphi+3m-3} + \beta_{6-4m}g_{\varphi+3m-2} + \beta_{4-4m}\ g_{\varphi+3m-1} + \beta_{2-4m}\ g_{\varphi+3m}.$$

We get all the unknowns $g_{2\varphi-2m}, g_{2\varphi-2m+1}, g_{2\varphi-2m+2}, \ldots, g_{2\varphi+2m-2}, g_{2\varphi+2m-1}, g_{2\varphi+2m}$. Now we substitute all these values in the four equations given in (5) and the result can be written in the following compact form

$$\begin{cases} g_{4\varphi-2} = \sum_{\lambda=-m+1}^{m} \beta_{4-4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{-2\alpha}\ g_{\varphi+\alpha+\lambda} \right) + \sum_{\lambda=-m+1}^{m} \beta_{2-4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{2+2\alpha}\ g_{\varphi+\alpha+\lambda} \right), \\ g_{4\varphi-1} = \sum_{\lambda=-m+1}^{m} \beta_{4\lambda-2} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{-2\alpha}\ g_{\varphi+\alpha+\lambda} \right) + \sum_{\lambda=-m+1}^{m} \beta_{4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{2+2\alpha}\ g_{\varphi+\alpha+\lambda} \right), \\ g_{4\varphi} = \sum_{\lambda=-m+1}^{m} \beta_{4-4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{2+2\alpha}\ g_{\varphi+\alpha+\lambda} \right) + \sum_{\lambda=-m+1}^{m} \beta_{2-4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{-2\alpha}\ g_{\varphi+\alpha+\lambda+1} \right), \\ g_{4\varphi+1} = \sum_{\lambda=-m+1}^{m} \beta_{4\lambda-2} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{2+2\alpha}\ g_{\varphi+\alpha+\lambda} \right) + \sum_{\lambda=-m+1}^{m} \beta_{4\lambda} \left(\sum_{\alpha=-2m}^{2m-1} \beta_{-2\alpha}\ g_{\varphi+\alpha+\lambda+1} \right). \end{cases} \tag{7}$$

This completes the proof. □

The following theorem presents a connection between the $4m$-point binary and the $(6m-1)$-point relaxed quaternary subdivision schemes.

Theorem 3: If $n = 2m$, then the subdivision equations given in (7) gives the four subdivision rules of the $(6m-1)$-point relaxed quaternary subdivision scheme whose coefficients of the control points in the subdivision rules are the non-linear combination of the coefficients of the control points of the $4m$-point binary subdivision scheme.

Proof: Now we add the subdivision level on the subdivision rules given in (7), Hence we get the following $(6m-1)$-point relaxed quaternary subdivision scheme

$$\begin{cases} g^{k+1}_{4\varphi-2} = \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4-4\lambda}\beta_{-2\alpha}\ g^k_{\varphi+\alpha+\lambda} + \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{2-4\lambda}\beta_{2+2\alpha}\ g^k_{\varphi+\alpha+\lambda}, \\ g^{k+1}_{4\varphi-1} = \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4\lambda-2}\beta_{-2\alpha}\ g^k_{\varphi+\alpha+\lambda} + \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4\lambda}\beta_{2+2\alpha}\ g^k_{\varphi+\alpha+\lambda}, \\ g^{k+1}_{4\varphi} = \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4-4\lambda}\beta_{2+2\alpha}\ g^k_{\varphi+\alpha+\lambda} + \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{2-4\lambda}\beta_{-2\alpha}\ g^k_{\varphi+\alpha+\lambda+1}, \\ g^{k+1}_{4\varphi+1} = \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4\lambda-2}\beta_{2+2\alpha}\ g^k_{\varphi+\alpha+\lambda} + \sum_{\lambda=-m+1}^{m} \sum_{\alpha=-2m}^{2m-1} \beta_{4\lambda}\beta_{-2\alpha}\ g^k_{\varphi+\alpha+\lambda+1}. \end{cases} \tag{8}$$

The mask coefficients of quaternary subdivision scheme (8) is the non-linear combination of the mask of the following $4m$-point binary subdivision scheme which we get by using $n = 2m$ in (2).

$$\begin{cases} g^{k+1}_{2\varphi-1} = \sum_{\lambda=-2m+1}^{2m} \beta_{2\lambda}\ g^k_{\varphi+\lambda-1}, \\ g^{k+1}_{2\varphi} = \sum_{\lambda=-2m+1}^{2m} \beta_{2-2\lambda}\ g^k_{\varphi+\lambda}, \end{cases} \tag{9}$$

□

The given theorems prove the generlized results about the odd n.

Theorem 4: If n is odd, that is $n = 2m+1 : m \in \mathbb{N}$, then the subdivision rules $g_{4\varphi-2}$ and $g_{4\varphi-1}$ in (4) are the linear combination of $6m+3$ control points $g_{\varphi-3m-1} \cdots g_{\varphi+3m+1}$, while the subdivision rules $g_{4\varphi}$ and $g_{4\varphi+1}$ in (4) are the linear combination of $6m+2$ control points $g_{\varphi-3m} \cdots g_{\varphi+3m+1}$.

Proof: When n is odd, we put $n = 2m+1$ in (4). That is

$$\begin{cases} g_{4\varphi-2} = \sum_{\lambda=-2m}^{2m+1} \beta_{2-2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi-1} = \sum_{\lambda=-2m}^{2m+1} \beta_{2\lambda}\ g_{2\varphi+\lambda-1}, \\ g_{4\varphi} = \sum_{\lambda=-2m}^{2m+1} \beta_{2-2\lambda}\ g_{2\varphi+\lambda}, \\ g_{4\varphi+1} = \sum_{\lambda=-2m}^{2m+1} \beta_{2\lambda}\ g_{2\varphi+\lambda}. \end{cases} \tag{10}$$

Now we have to find out values of $g_{2\varphi-2m-1}$, $g_{2\varphi-2m}$, $g_{2\varphi-2m+1}$, …, $g_{2\varphi+2m}$, $g_{2\varphi+2m+1}$. For this, first we evaluate (3) for $n = 2m+1$ and then by increasing or decreasing the subscript, we get the required unknowns.

$$\begin{cases} g_{2\varphi-1} = \beta_{-4m}\ g_{\varphi-2m-1} + \beta_{2-4m}\ g_{\varphi-2m} + \beta_{4-4m} \\ \quad \times\ g_{\varphi-2m+1} + \beta_{6-4m}\ g_{\varphi-2m+2} + \beta_{8-4m} \\ \quad \times g_{\varphi-2m+3} + \beta_{10-4m}\ g_{\varphi-2m+4} + \beta_{12-4m} \\ \quad \times g_{\varphi-2m+5} + \ldots + \beta_{4m-8} g_{\varphi+2m-5} + \beta_{4m-6} \\ \quad \times g_{\varphi+2m-4} + \beta_{4m-4}\ g_{\varphi+2m-3} + \beta_{4m-2} \\ \quad \times g_{\varphi+2m-2} + \beta_{4m} g_{\varphi+2m-1} + \beta_{4m+2} g_{\varphi+2m}, \\ g_{2\varphi} = \beta_{4m+2}\ g_{\varphi-2m} + \beta_{4m}\ g_{\varphi-2m+1} + \beta_{4m-2} \\ \quad \times g_{\varphi-2m+2} + \beta_{4m-4}\ g_{\varphi-2m+3} + \beta_{4m-6} \\ \quad \times g_{\varphi-2m+4} + \beta_{4m-8}\ g_{\varphi-2m+5} + \beta_{4m-10} \\ \quad \times\ g_{\varphi-2m+6} + \ldots + \beta_{10-4m}\ g_{\varphi+2m-4} \\ \quad + \beta_{8-4m}\ g_{\varphi+2m-3} + \beta_{6-4m}\ g_{\varphi+2m-2} \\ \quad + \beta_{4-4m}\ g_{\varphi+2m-1} + \beta_{2-4m}\ g_{\varphi+2m} \\ \quad + \beta_{-4m}\ g_{\varphi+2m+1}. \end{cases} \tag{11}$$

Now we replace φ by $\varphi - m$ in the first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi-2m-1} = {} & \beta_{-4m}\ g_{\varphi-3m-1} + \beta_{2-4m}\ g_{\varphi-3m} + \beta_{4-4m} \\ & \times g_{\varphi-3m+1} + \beta_{6-4m}\ g_{\varphi-3m+2} + \beta_{8-4m}\ g_{\varphi-3m+3} \\ & + \beta_{10-4m} g_{\varphi-3m+4} + \beta_{12-4m} g_{\varphi-3m+5} + \beta_{14-4m} \\ & \times g_{\varphi-3m+6} + \ldots + \beta_{4m-8}\ g_{\varphi+m-5} + \beta_{4m-6} \\ & \times g_{\varphi+m-4} + \beta_{4m-4}\ g_{\varphi+m-3} + \beta_{4m-2}\ g_{\varphi+m-2} \\ & + \beta_{4m}\ g_{\varphi+m-1} + \beta_{4m+2} g_{\varphi+m}, \\ g_{2\varphi-2m} = {} & \beta_{4m+2}\ g_{\varphi-3m} + \beta_{4m} g_{\varphi-3m+1} + \beta_{4m-2} g_{\varphi-3m+2} \\ & + \beta_{4m-4}\ g_{\varphi-3m+3} + \beta_{4m-6}\ g_{\varphi-3m+4} + \beta_{4m-8} \\ & \times g_{\varphi-3m+5} + \beta_{4m-10}\ g_{\varphi-3m+6} + \ldots + \beta_{10-4m} \\ & \times g_{\varphi+m-4} + \beta_{8-4m}\ g_{\varphi+m-3} + \beta_{6-4m}\ g_{\varphi+m-2} \\ & + \beta_{4-4m} g_{\varphi+m-1} + \beta_{2-4m} g_{\varphi+m} + \beta_{-4m} g_{\varphi+m+1}. \end{aligned}$$

Now we replace φ by $\varphi - m + 1$ in the first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi-2m+1} = {} & \beta_{-4m} g_{\varphi-3m} + \beta_{2-4m} g_{\varphi-3m+1} + \beta_{4-4m} g_{\varphi-3m+2} \\ & + \beta_{6-4m} g_{\varphi-3m+3} + \beta_{8-4m}\ g_{\varphi-3m+4} + \beta_{10-4m} \\ & \times g_{\varphi-3m+5} + \beta_{12-4m} g_{\varphi-3m+6} + \ldots + \beta_{4m-8} \\ & \times g_{\varphi+m-4} + \beta_{4m-6} g_{\varphi+m-3} + \beta_{4m-4}\ g_{\varphi+m-2} \\ & + \beta_{4m-2} g_{\varphi+m-1} + \beta_{4m}\ g_{\varphi+m} + \beta_{4m+2}\ g_{\varphi+m+1}, \\ g_{2\varphi-2m+2} = {} & \beta_{4m+2} g_{\varphi-3m+1} + \beta_{4m} g_{\varphi-3m+2} + \beta_{4m-2} g_{\varphi-3m+3} \\ & + \beta_{4m-4}\ g_{\varphi-3m+4} + \beta_{4m-6}\ g_{\varphi-3m+5} + \beta_{4m-8} \\ & \times g_{\varphi-3m+6} + \beta_{4m-10}\ g_{\varphi-3m+7} + \ldots + \beta_{10-4m} \\ & \times g_{\varphi+m-3} + \beta_{8-4m}\ g_{\varphi+m-2} + \beta_{6-4m}\ g_{\varphi+m-1} \\ & + \beta_{4-4m} g_{\varphi+m} + \beta_{2-4m} g_{\varphi+m+1} + \beta_{-4m} g_{\varphi+m+2}. \end{aligned}$$

Now we replace φ by $\varphi - m + 2$ in the first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi-2m+3} = {} & \beta_{-4m} g_{\varphi-3m+1} + \beta_{2-4m} g_{\varphi-3m+2} + \beta_{4-4m} \\ & \times g_{\varphi-3m+3} + \beta_{6-4m}\ g_{\varphi-3m+4} + \beta_{8-4m} \\ & \times g_{\varphi-3m+5} + \beta_{10-4m}\ g_{\varphi-3m+6} + \beta_{12-4m} \\ & \times g_{\varphi-3m+7} + \ldots + \beta_{4m-8}\ g_{\varphi+m-3} + \beta_{4m-6} \\ & \times g_{\varphi+m-2} + \beta_{4m-4}\ g_{\varphi+m-1} + \beta_{4m-2} g_{\varphi+m} \\ & + \beta_{4m}\ g_{\varphi+m+1} + \beta_{4m+2}\ g_{\varphi+m+2}, \\ g_{2\varphi-2m+4} = {} & \beta_{4m+2}\ g_{\varphi-3m+2} + \beta_{4m}\ g_{\varphi-3m+3} + \beta_{4m-2} \\ & \times g_{\varphi-3m+4} + \beta_{4m-4} g_{\varphi-3m+5} + \beta_{4m-6} g_{\varphi-3m+6} \\ & + \beta_{4m-8} g_{\varphi-3m+7} + \beta_{4m-10} g_{\varphi-3m+8} + \ldots \\ & + \beta_{10-4m}\ g_{\varphi+m-2} + \beta_{8-4m}\ g_{\varphi+m-1} + \beta_{6-4m} \\ & \times g_{\varphi+m} + \beta_{4-4m}\ g_{\varphi+m+1} + \beta_{2-4m}\ g_{\varphi+m+2} \\ & + \beta_{-4m}\ g_{\varphi+m+3}. \end{aligned}$$

$$\vdots \qquad\qquad \vdots$$

continuiting this process, we replace φ by $\varphi + m - 2$ in first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi+2m-5} = {} & \beta_{-4m}\ g_{\varphi-m-3} + \beta_{2-4m}\ g_{\varphi-m-2} + \beta_{4-4m} \\ & \times g_{\varphi-m-1} + \beta_{6-4m}\ g_{\varphi-m} + \beta_{8-4m}\ g_{\varphi-m+1} \\ & + \beta_{10-4m}\ g_{\varphi-m+2} + \beta_{12-4m}\ g_{\varphi-m+3} + \ldots \\ & + \beta_{4m-8}\ g_{\varphi+3m-7} + \beta_{4m-6}\ g_{\varphi+3m-6} + \beta_{4m-4} \\ & \times g_{\varphi+3m-5} + \beta_{4m-2} g_{\varphi+3m-4} + \beta_{4m} g_{\varphi+3m-3} \\ & + \beta_{4m+2}\ g_{\varphi+3m-2}, \\ g_{2\varphi+2m-4} = {} & \beta_{4m+2}\ g_{\varphi-m-2} + \beta_{4m} g_{\varphi-m-1} + \beta_{4m-2} g_{\varphi-m} \\ & + \beta_{4m-4}\ g_{\varphi-m+1} + \beta_{4m-6}\ g_{\varphi-m+2} + \beta_{4m-8} \\ & \times g_{\varphi-m+3} + \beta_{4m-10}\ g_{\varphi-m+4} + \ldots + \beta_{10-4m} \\ & \times\ g_{\varphi+3m-6} + \beta_{8-4m}\ g_{\varphi+3m-5} + \beta_{6-4m} \\ & \times g_{\varphi+3m-4} + \beta_{4-4m} g_{\varphi+3m-3} + \beta_{2-4m} g_{\varphi+3m-2} \\ & + \beta_{-4m}\ g_{\varphi+3m-1}. \end{aligned}$$

Now we replace φ by $\varphi + m - 1$ in the first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi+2m-3} = {} & \beta_{-4m}\ g_{\varphi-m-2} + \beta_{2-4m}\ g_{\varphi-m-1} + \beta_{4-4m} \\ & \times g_{\varphi-m} + \beta_{6-4m}\ g_{\varphi-m+1} + \beta_{8-4m}\ g_{\varphi-m+2} \\ & + \beta_{10-4m}\ g_{\varphi-m+3} + \beta_{12-4m}\ g_{\varphi-m+4} + \ldots \\ & + \beta_{4m-8}\ g_{\varphi+3m-6} + \beta_{4m-6}\ g_{\varphi+3m-5} + \beta_{4m-4} \\ & \times g_{\varphi+3m-4} + \beta_{4m-2} g_{\varphi+3m-3} + \beta_{4m} g_{\varphi+3m-2} \\ & + \beta_{4m+2}\ g_{\varphi+3m-1}, \\ g_{2\varphi+2m-2} = {} & \beta_{4m+2}\ g_{\varphi-m-1} + \beta_{4m} g_{\varphi-m} + \beta_{4m-2} g_{\varphi-m+1} \\ & + \beta_{4m-4}\ g_{\varphi-m+2} + \beta_{4m-6}\ g_{\varphi-m+3} + \beta_{4m-8} \\ & \times g_{\varphi-m+4} + \beta_{4m-10}\ g_{\varphi-m+5} + \ldots + \beta_{10-4m} \\ & \times g_{\varphi+3m-5} + \beta_{8-4m}\ g_{\varphi+3m-4} + \beta_{6-4m} \\ & \times g_{\varphi+3m-3} + \beta_{4-4m} g_{\varphi+3m-2} + \beta_{2-4m} g_{\varphi+3m-1} \\ & + \beta_{-4m}\ g_{\varphi+3m}. \end{aligned}$$

Now we replace φ by $\varphi + m$ in the first and second rules of (11), we get

$$\begin{aligned} g_{2\varphi+2m-1} &= \beta_{-4m}\ g_{\varphi-m-1} + \beta_{2-4m}\ g_{\varphi-m} + \beta_{4-4m} \\ &\times g_{\varphi-m+1} + \beta_{6-4m} g_{\varphi-m+2} + \beta_{8-4m} g_{\varphi-m+3} \\ &+ \beta_{10-4m}\ g_{\varphi-m+4} + \beta_{12-4m}\ g_{\varphi-m+5} + \dots \\ &+ \beta_{4m-8} g_{\varphi+3m-5} + \beta_{4m-6} g_{\varphi+3m-4} + \beta_{4m-4} \\ &\times g_{\varphi+3m-3} + \beta_{4m-2} g_{\varphi+3m-2} + \beta_{4m} g_{\varphi+3m-1} \\ &+ \beta_{4m+2}\ g_{\varphi+3m}, \\ g_{2\varphi+2m} &= \beta_{4m+2}\ g_{\varphi-m} + \beta_{4m} g_{\varphi-m+1} + \beta_{4m-2} g_{\varphi-m+2} \\ &+ \beta_{4m-4}\ g_{\varphi-m+3} + \beta_{4m-6}\ g_{\varphi-m+4} + \beta_{4m-8} \\ &\times g_{\varphi-m+5} + \beta_{4m-10}\ g_{\varphi-m+6} + \dots + \beta_{10-4m} \\ &\times g_{\varphi+3m-4} + \beta_{8-4m}\ g_{\varphi+3m-3} + \beta_{6-4m} \\ &\times g_{\varphi+3m-2} + \beta_{4-4m}\ g_{\varphi+3m-1} + \beta_{2-4m}\ g_{\varphi+3m} \\ &+ \beta_{-4m}\ g_{\varphi+3m+1}. \end{aligned}$$

Now we replace φ by $\varphi+m+1$ in the first rule of (11), we get

$$\begin{aligned} g_{2\varphi+2m+1} &= \beta_{-4m}\ g_{\varphi-m} + \beta_{2-4m}\ g_{\varphi-m+1} + \beta_{4-4m} \\ &\times g_{\varphi-m+2} + \beta_{6-4m}\ g_{\varphi-m+3} + \beta_{8-4m} g_{\varphi-m+4} \\ &+ \beta_{10-4m}\ g_{\varphi-m+5} + \beta_{12-4m}\ g_{\varphi-m+6} + \dots \\ &+ \beta_{4m-8} g_{\varphi+3m-4} + \beta_{4m-6} g_{\varphi+3m-3} + \beta_{4m-4} \\ &\times g_{\varphi+3m-2} + \beta_{4m-2}\ g_{\varphi+3m-1} + \beta_{4m}\ g_{\varphi+3m} \\ &+ \beta_{4m+2}\ g_{\varphi+3m+1}. \end{aligned}$$

Now, we get all the unknowns $g_{2\varphi-2m-1}, g_{2\varphi-2m}, g_{2\varphi-2m+1}, \dots, g_{2\varphi+2m}, g_{2\varphi+2m+1}$. Further, we substitute these in the set of equations (10), which in the short form can be written as

$$\begin{cases} g_{4\varphi-2} &= \sum_{\lambda=-m}^{m} \beta_{2-4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2\alpha}\ g_{\varphi+\alpha+\lambda-1} \right) \\ & \quad + \sum_{\lambda=-m}^{m} \beta_{-4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2-2\alpha}\ g_{\varphi+\alpha+\lambda} \right), \\ g_{4\varphi-1} &= \sum_{\lambda=-m}^{m} \beta_{4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2\alpha}\ g_{\varphi+\alpha+\lambda-1} \right) \\ & \quad + \sum_{\lambda=-m}^{m} \beta_{2+4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2-2\alpha}\ g_{\varphi+\alpha+\lambda} \right), \\ g_{4\varphi} &= \sum_{\lambda=-m}^{m} \beta_{2-4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2-2\alpha}\ g_{\varphi+\alpha+\lambda} \right) \\ & \quad + \sum_{\lambda=-m}^{m} \beta_{-4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2\alpha}\ g_{\varphi+\alpha+\lambda} \right), \\ g_{4\varphi+1} &= \sum_{\lambda=-m}^{m} \beta_{4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2-2\alpha}\ g_{\varphi+\alpha+\lambda} \right) \\ & \quad + \sum_{\lambda=-m}^{m} \beta_{2+4\lambda} \left(\sum_{\alpha=-2m}^{2m+1} \beta_{2\alpha}\ g_{\varphi+\alpha+\lambda} \right). \end{cases} \tag{12}$$

Which completes the required result. □

The following theorem derives a relation between the $(4m + 2)$-point binary and the $(6m + 2)$-point relaxed quaternary subdivision schemes.

Theorem 5: If $n = 2m + 1$, then the subdivision equations given in (12) gives the four subdivision rules of the $(6m+2)$-point relaxed quaternary subdivision scheme whose coefficients of the control points in the subdivision rules are the non-linear combination of the coefficients of the control points of the $(4m + 2)$-point binary subdivision scheme.

Proof: Now we add the subdivision level on the subdivision rules given in (12), Hence we get the following $(6m + 2)$-point relaxed quaternary subdivision scheme

$$\begin{cases} g^{k+1}_{4\varphi-2} = \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{2-4\lambda}\beta_{2\alpha}\ g^k_{\varphi+\alpha+\lambda-1} \\ \quad + \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{-4\lambda}\beta_{2-2\alpha}\ g^k_{\varphi+\alpha+\lambda}, \\ g^{k+1}_{4\varphi-1} = \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{4\lambda}\beta_{2\alpha}\ g^k_{\varphi+\alpha+\lambda-1} \\ \quad + \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{2+4\lambda}\beta_{2-2\alpha}\ g^k_{\varphi+\alpha+\lambda}, \\ g^{k+1}_{4\varphi} = \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{2-4\lambda}\beta_{2-2\alpha}\ g^k_{\varphi+\alpha+\lambda} \\ \quad + \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{-4\lambda}\beta_{2\alpha}\ g^k_{\varphi+\alpha+\lambda}, \\ g^{k+1}_{4\varphi+1} = \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{4\lambda}\beta_{2-2\alpha}\ g^k_{\varphi+\alpha+\lambda} \\ \quad + \sum_{\lambda=-m}^{m} \sum_{\alpha=-2m}^{2m+1} \beta_{2+4\lambda}\beta_{2\alpha}\ g^k_{\varphi+\alpha+\lambda}. \end{cases} \tag{13}$$

The mask coefficients of quaternary subdivision scheme (13) is the non-linear combination of the mask of the following $(4m + 2)$-point binary subdivision scheme which we get by using $n = 2m + 1$ in (2).

$$\begin{cases} g^{k+1}_{2\varphi-1} = \sum_{\lambda=-2m}^{2m+1} \beta_{2\lambda}\ g^k_{\varphi+\lambda-1}, \\ g^{k+1}_{2\varphi} = \sum_{\lambda=-2m}^{2m+1} \beta_{2-2\lambda}\ g^k_{\varphi+\lambda}, \end{cases} \tag{14}$$

Hence proved. □

In the next section, we will validate the results of Theorem 3 and Theorem 5.

III. APPLICATIONS OF THE PRESENTED TECHNIQUES

In this section, we implement and validate the results, which are proved in Theorem 3 and Theorem 5 of Section II, to the known even-point binary approximating subdivision schemes. We also inspects the graphical results of both type of schemes using the same initial data.We use non-parametric binary subdivision schemes in the Corollaries 6-13, but the given procedure can be applied on all the parametric as well as the non-parametric linear even-point binary subdivision schemes. The first three corollaries are the applications of Theorem 3 whereas the next four are the applications of Theorem 5.

Corollary 6: We expand the binary subdivision scheme which is defined in (9) for $m = 1$, thus we get

$$\begin{cases} g^{k+1}_{2\varphi-1} = \beta_{-2}\ g^k_{\varphi-2} + \beta_0\ g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_4\ g^k_{\varphi+1}, \\ g^{k+1}_{2\varphi} = \beta_4\ g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_0\ g^k_{\varphi+1} + \beta_{-2}\ g^k_{\varphi+2}. \end{cases} \tag{15}$$

To get the values of $\beta_{-2}, \beta_0, \beta_2, \beta_4$, we compare the general form of the 4-point binary scheme (15) with the 4-point scheme defined by [30], hence we get

$$\beta_{-2} = \frac{1}{384}, \quad \beta_0 = \frac{121}{384}, \quad \beta_2 = \frac{235}{384}, \quad \beta_4 = \frac{9}{128}. \tag{16}$$

Now by using the mask (16), we can get the mask/coefficients of the quaternary subdivision scheme. Hence by expanding (8) for $m = 1$, we get

$$\begin{aligned} g^{k+1}_{4\varphi-2} &= (\beta_4^2 + \beta_2\beta_{-2})g^k_{\varphi-2} + (\beta_4\beta_2 + \beta_0\beta_4 + \beta_2\beta_0 \\ &\quad + \beta_{-2}^2)g^k_{\varphi-1} + (\beta_4\beta_0 + \beta_0\beta_2 + \beta_2^2 + \beta_{-2}\beta_0)g^k_{\varphi} \\ &\quad + (\beta_4\beta_{-2} + \beta_0^2 + \beta_2\beta_4 + \beta_{-2}\beta_2)g^k_{\varphi+1} + (\beta_0 \\ &\quad \times \beta_{-2} + \beta_{-2}\beta_4)g^k_{\varphi+2}, \\ g^{k+1}_{4\varphi-1} &= (\beta_{-2}\beta_4 + \beta_0\beta_{-2})g^k_{\varphi-2} + (\beta_{-2}\beta_2 + \beta_2\beta_4 + \beta_0^2 \\ &\quad + \beta_4\beta_{-2})g^k_{\varphi-1} + (\beta_{-2}\beta_0 + \beta_2^2 + \beta_0\beta_2 + \beta_4\beta_0) \\ &\quad \times g^k_{\varphi} + (\beta_{-2}^2 + \beta_2\beta_0 + \beta_0\beta_4 + \beta_4\beta_2)g^k_{\varphi+1} + (\beta_2 \\ &\quad \times \beta_{-2} + \beta_4^2)g^k_{\varphi+2}, \\ g^{k+1}_{4\varphi} &= (\beta_4\beta_{-2})g^k_{\varphi-2} + (\beta_4\beta_0 + \beta_0\beta_{-2} + \beta_2\beta_4)g^k_{\varphi-1} \\ &\quad + (\beta_4\beta_2 + \beta_0^2 + \beta_2^2 + \beta_{-2}\beta_4)g^k_{\varphi} + (\beta_4^2 + \beta_0\beta_2 + \beta_2 \\ &\quad \times \beta_0 + \beta_{-2}\beta_2)g^k_{\varphi+1} + (\beta_0\beta_4 + \beta_2\beta_{-2} + \beta_{-2}\beta_0) \\ &\quad \times g^k_{\varphi+2} + (\beta_{-2}^2)g^k_{\varphi+3}, \\ g^{k+1}_{4\varphi+1} &= (\beta_{-2}^2)g^k_{\varphi-2} + (\beta_{-2}\beta_0 + \beta_2\beta_{-2} + \beta_0\beta_4)g^k_{\varphi-1} \\ &\quad + (\beta_{-2}\beta_2 + \beta_2\beta_0 + \beta_0\beta_2 + \beta_4^2)g^k_{\varphi} + (\beta_{-2}\beta_4 + \beta_2^2 \\ &\quad + \beta_0^2 + \beta_4\beta_2)g^k_{\varphi+1} + (\beta_2\beta_4 + \beta_0\beta_{-2} + \beta_4\beta_0) \\ &\quad \times g^k_{\varphi+2} + (\beta_4\beta_{-2})g^k_{\varphi+3}. \end{aligned}$$

By using the values of β_{-2}, β_0, β_2 and β_4 from (16) in above, we get

$$\begin{cases} g^{k+1}_{4\varphi-2} = \hat{A}_1 g^k_{\varphi-2} + \hat{A}_2 g^k_{\varphi-1} + \hat{A}_3 g^k_{\varphi} + \hat{A}_4 g^k_{\varphi+1} \\ \qquad + \hat{A}_5 g^k_{\varphi+2}, \\ g^{k+1}_{4\varphi-1} = \hat{A}_5 g^k_{\varphi-2} + \hat{A}_4 g^k_{\varphi-1} + \hat{A}_3 g^k_{\varphi} + \hat{A}_2 g^k_{\varphi+1} \\ \qquad + \hat{A}_1 g^k_{\varphi+2}, \\ g^{k+1}_{4\varphi} = \hat{B}_1 g^k_{\varphi-2} + \hat{B}_2 g^k_{\varphi-1} + \hat{B}_3 g^k_{\varphi} + \hat{B}_4 g^k_{\varphi+1} \\ \qquad + \hat{B}_5 g^k_{\varphi+2} + \hat{B}_6 g^k_{\varphi+3}, \\ g^{k+1}_{4\varphi+1} = \hat{B}_6 g^k_{\varphi-2} + \hat{B}_5 g^k_{\varphi-1} + \hat{B}_4 g^k_{\varphi} + \hat{B}_3 g^k_{\varphi+1} \\ \qquad + \hat{B}_2 g^k_{\varphi+2} + \hat{B}_1 g^k_{\varphi+3}, \end{cases} \tag{17}$$

where

$$\begin{cases} \hat{A}_1 = \frac{241}{36864}, \hat{A}_2 = \frac{1189}{4608}, \hat{A}_3 = \frac{1209}{2048}, \hat{A}_4 = \frac{83}{576}, \\ \hat{A}_5 = \frac{37}{36864}, \hat{B}_1 = \frac{3}{16384}, \hat{B}_2 = \frac{9733}{147456}, \\ \hat{B}_3 = \frac{38119}{73728}, \hat{B}_4 = \frac{3213}{8192}, \hat{B}_5 = \frac{3623}{147456}, \hat{B}_6 = \frac{1}{147456} \end{cases} \tag{18}$$

Which is the 5-point relaxed quaternary subdivision scheme. The mask/coefficients of this quaternary subdivision scheme (17) is just the non-linear combination of the mask of the binary subdivision scheme (15).

The graphical inspection and comparison of the binary subdivision scheme (15) and the quaternary subdivision scheme (17) after one and two subdivision steps is given in Figure 1. This figure clearly shows that the quaternary subdivision scheme smooths the model more efficiently as compare to the binary subdivision scheme.

Remark 7: In captions of the Figures 1-7, SS, BSS and QSS denote the Subdivision Step, Binary Subdivision Scheme and the Quaternary Subdivision Scheme respectively. Moreover, in Figures 1-7 red solid lines represent the initial polygons, blue solid lines show the curves fitted by the binary and the quaternary subdivision schemes after one subdivision level, while the black solid lines show the curves fitted by the binary and quaternary subdivision schemes after two subdivision steps.

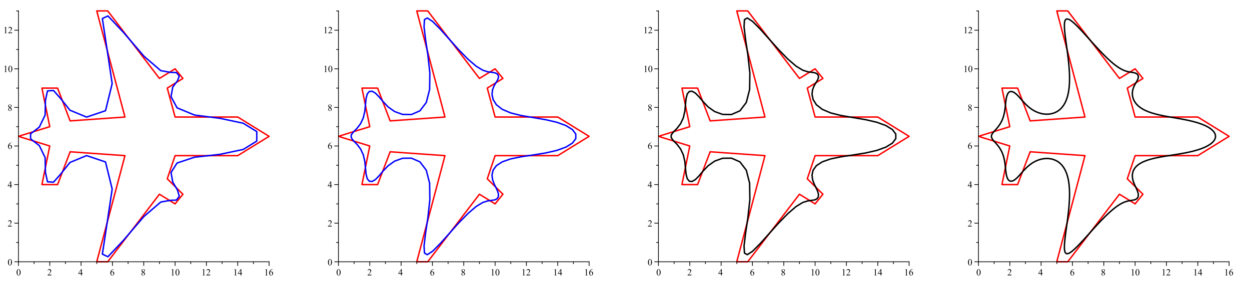
Corollary 8: This corollary is the application of Theorem 3 for $m = 2$. The binary subdivision scheme (9) for $m = 2$ is:

$$\begin{cases} g^{k+1}_{2\varphi-1} = \beta_{-6}\ g^k_{\varphi-4} + \beta_{-4}\ g^k_{\varphi-3} + \beta_{-2}\ g^k_{\varphi-2} \\ \quad + \beta_0\ g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_4\ g^k_{\varphi+1} \\ \quad + \beta_6\ g^k_{\varphi+2} + \beta_8\ g^k_{\varphi+3}, \\ g^{k+1}_{2\varphi} = \beta_8\ g^k_{\varphi-3} + \beta_6\ g^k_{\varphi-2} + \beta_4\ g^k_{\varphi-1} \\ \quad + \beta_2\ g^k_{\varphi} + \beta_0\ g^k_{\varphi+1} + \beta_{-2}\ g^k_{\varphi+2} \\ \quad + \beta_{-4}\ g^k_{\varphi+3} + \beta_{-6}\ g^k_{\varphi+4}. \end{cases} \tag{19}$$

(19) is the general form of 8-point binary subdivision scheme, the coefficients $\beta_{-6}, \beta_{-4}, \ldots, \beta_6, \beta_8$ of which can be get by any of the known 8-point binary subdivision scheme. Therefore, in order to get coefficients we compare the scheme (19) with the 8-point scheme presented by [30], so we get

$$\begin{cases} \beta_{-6} = \frac{1}{82575360}, \beta_{-4} = \frac{26039}{27525120}, \beta_{-2} = \frac{1385999}{27525120}, \\ \beta_0 = \frac{26672209}{82575360}, \beta_2 = \frac{4210971}{9175040}, \beta_4 = \frac{1440007}{9175040}, \\ \beta_6 = \frac{806047}{82575360}, \quad \beta_8 = \frac{243}{9175040}. \end{cases} \tag{20}$$

If we put $m = 2$ in (8), we get the 12-point relaxed quaternary subdivision scheme whose mask is the non-linear



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 1. Curves generated by the binary and quaternary subdivision schemes (15) and (17) respectively.

combination of the mask $\beta_{-6}, \beta_{-4}, \ldots, \beta_6, \beta_8$ *of the* 8-*point binary subdivision scheme. Therefore, by using (20), we get the mask of the following quternary subdivision scheme*

$$\begin{cases} g^{k+1}_{4\varphi-2} = \breve{A}_1 g^k_{\varphi-5} + \breve{A}_2 g^k_{\varphi-4} + \breve{A}_3 g^k_{\varphi-3} + \breve{A}_4 g^k_{\varphi-2} \\ \quad + \breve{A}_5 g^k_{\varphi-1} + \breve{A}_6 g^k_{\varphi} + \breve{A}_7 g^k_{\varphi+1} + \breve{A}_8 g^k_{\varphi+2} \\ \quad + \breve{A}_9 g^k_{\varphi+3} + \breve{A}_{10} g^k_{\varphi+4} + \breve{A}_{11} g^k_{\varphi+5}, \\ g^{k+1}_{4\varphi-1} = \breve{A}_{11} g^k_{\varphi-5} + \breve{A}_{10} g^k_{\varphi-4} + \breve{A}_9 g^k_{\varphi-3} + \breve{A}_8 \\ \quad \times g^k_{\varphi-2} + \breve{A}_7 g^k_{\varphi-1} + \breve{A}_6 g^k_{\varphi} + \breve{A}_5 g^k_{\varphi+1} \\ \quad + \breve{A}_4 g^k_{\varphi+2} + \breve{A}_3 g^k_{\varphi+3} + \breve{A}_2 g^k_{\varphi+4} \\ \quad + \breve{A}_1 g^k_{\varphi+5}, \\ g^{k+1}_{4\varphi} = \breve{B}_1 g^k_{\varphi-5} + \breve{B}_2 g^k_{\varphi-4} + \breve{B}_3 g^k_{\varphi-3} + \breve{B}_4 g^k_{\varphi-2} \\ \quad + \breve{B}_5 g^k_{\varphi-1} + \breve{B}_6 g^k_{\varphi} + \breve{B}_7 g^k_{\varphi+1} + \breve{B}_8 g^k_{\varphi+2} \\ \quad + \breve{B}_9 g^k_{\varphi+3} + \breve{B}_{10} g^k_{\varphi+4} + \breve{B}_{11} g^k_{\varphi+5} \\ \quad + \breve{B}_{12} g^k_{\varphi+6}, \\ g^{k+1}_{4\varphi+1} = \breve{B}_{12} g^k_{\varphi-5} + \breve{B}_{11} g^k_{\varphi-4} + \breve{B}_{10} g^k_{\varphi-3} + \breve{B}_9 \\ \quad \times g^k_{\varphi-2} + \breve{B}_8 g^k_{\varphi-1} + \breve{B}_7 g^k_{\varphi} + \breve{B}_6 g^k_{\varphi+1} \\ \quad + \breve{B}_5 g^k_{\varphi+2} + \breve{B}_4 g^k_{\varphi+3} + \breve{B}_3 g^k_{\varphi+4} + \breve{B}_2 \\ \quad \times g^k_{\varphi+5} + \breve{B}_1 g^k_{\varphi+6}, \end{cases} \tag{21}$$

where

$$\begin{cases} \breve{A}_1 = \dfrac{698627}{852336259891200}, \quad \breve{A}_2 = \dfrac{3879598117}{284112086630400}, \\ \breve{A}_3 = \dfrac{350941180003}{142056043315200}, \quad \breve{A}_4 = \dfrac{36602385889}{676457349120}, \\ \breve{A}_5 = \dfrac{5641800724981}{20293720473600}, \quad \breve{A}_6 = \dfrac{247891317863}{579820584960}, \\ \breve{A}_7 = \dfrac{527402518309}{2536715059200}, \quad \breve{A}_8 = \dfrac{2067049873661}{71028021657600}, \\ \breve{A}_9 = \dfrac{48690122269}{56822417326080}, \quad \breve{A}_{10} = \dfrac{1902910423}{852336259891200}, \end{cases} \tag{22}$$

$$\begin{cases} \breve{A}_{11} = \dfrac{239}{20293720473600}\, \breve{B}_1 = \dfrac{27}{84181359001600}, \\ \breve{B}_2 = \dfrac{30898837}{108233175859200}, \ \breve{B}_3 = \dfrac{1754117761421}{6818690079129600}, \\ \breve{B}_4 = \dfrac{32344488846199}{2272896693043200}, \ \breve{B}_5 = \dfrac{163622535291293}{1136448346521600}, \\ \breve{B}_6 = \dfrac{12926815750607}{32469952757760}, \ \breve{B}_7 = \dfrac{56015931444329}{162349763788800}, \\ \breve{B}_8 = \dfrac{104604880235159}{1136448346521600}, \ \breve{B}_9 = \dfrac{75466139809}{12025908428800}, \\ \breve{B}_{10} = \dfrac{148716800989}{2272896693043200}, \ \breve{B}_{11} = \dfrac{58359331}{2272896693043200}, \\ \breve{B}_{12} = \dfrac{1}{6818690079129600}. \end{cases} \tag{23}$$

Figure 2 shows the models fitted by the binary subdivision scheme (19) and the quaternary subdivision scheme (21) after one and two subdivision steps. Again this graphical inspection shows the superiority of the quaternary subdivision scheme over the binary subdivision scheme.

Corollary 9: In this corollary, we use the even-point binary subdivision scheme (9) and expand it when $m = 3$. *As the result we get the following* 12-*point binary subdivision scheme*:

$$\begin{cases} g^{k+1}_{2\varphi-1} = \beta_{-10}\ g^k_{\varphi-6} + \beta_{-8}\ g^k_{\varphi-5} + \beta_{-6}\ g^k_{\varphi-4} \\ \quad + \beta_{-4}\ g^k_{\varphi-3} + \beta_{-2}\ g^k_{\varphi-2} + \beta_0\ g^k_{\varphi-1} \\ \quad + \beta_2\ g^k_{\varphi} + \beta_4\ g^k_{\varphi+1} + \beta_6\ g^k_{\varphi+2} + \beta_8 \\ \quad \times g^k_{\varphi+3} + \beta_{10}\ g^k_{\varphi+4} + \beta_{12}\ g^k_{\varphi+5}, \\ g^{k+1}_{2\varphi} = \beta_{12}\ g^k_{\varphi-5} + \beta_{10}\ g^k_{\varphi-4} + \beta_8\ g^k_{\varphi-3} + \beta_6 \\ \quad \times g^k_{\varphi-2} + \beta_4\ g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_0\ g^k_{\varphi+1} \\ \quad + \beta_{-2}\ g^k_{\varphi+2} + \beta_{-4}\ g^k_{\varphi+3} + \beta_{-6}\ g^k_{\varphi+4} \\ \quad + \beta_{-8}\ g^k_{\varphi+5} + \beta_{-10}\ g^k_{\varphi+6}. \end{cases} \tag{24}$$

A 12-*point binary subdivision scheme which we get by the algorithm presented by [30] gives the coefficients* $\beta_{-10}, \beta_{-8}, \ldots, \beta_{10}, \beta_{12}$ *of this scheme. Hence the coeffi-*

cients are:

$$\begin{cases} \beta_{-10} = \frac{1}{167423193907200}, \quad \beta_{-8} = \frac{48828113}{167423193907200}, \\ \beta_{-6} = \frac{410601629}{2232309252096}, \quad \beta_{-4} = \frac{52548530917}{6200859033600}, \\ \beta_{-2} = \frac{2471063221063}{27903865651200}, \quad \beta_{0} = \frac{1680588922139}{5580773130240}, \\ \beta_{2} = \frac{2134020225233}{5580773130240}, \quad \beta_{4} = \frac{69086299223}{372051542016}, \\ \beta_{6} = \frac{1785493468247}{55807731302400}, \quad \beta_{8} = \frac{261595441397}{167423193907200}, \\ \beta_{10} = \frac{1975200979}{167423193907200}, \quad \beta_{12} = \frac{2187}{2066953011200}. \end{cases} \tag{25}$$

Now first we simplify (8) for $m = 3$ *and then substitute these 12 values from (25) into (8), thus we get the 17-point relaxed quaternary approximating subdivision scheme, that is*

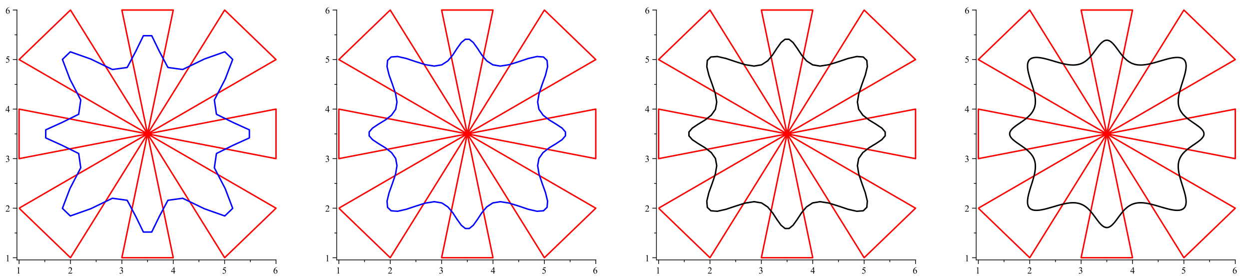
$$\begin{cases} g^{k+1}_{4\varphi-2} = C_1 g^k_{\varphi-8} + C_2 g^k_{\varphi-7} + C_3 g^k_{\varphi-6} + C_4 g^k_{\varphi-5} \\ \quad + C_5 g^k_{\varphi-4} + C_6 g^k_{\varphi-3} + C_7 g^k_{\varphi-2} + C_8 g^k_{\varphi-1} \\ \quad + C_9 g^k_{\varphi} + C_{10} g^k_{\varphi+1} + C_{11} g^k_{\varphi+2} + C_{12} g^k_{\varphi+3} \\ \quad + C_{13} g^k_{\varphi+4} + C_{14} g^k_{\varphi+5} + C_{15} g^k_{\varphi+6} + C_{16} g^k_{\varphi+7} \\ \quad + C_{17} g^k_{\varphi+8}, \\ g^{k+1}_{4\varphi-1} = C_{17} g^k_{\varphi-8} + C_{16} g^k_{\varphi-7} + C_{15} g^k_{\varphi-6} + C_{14} g^k_{\varphi-5} \\ \quad + C_{13} g^k_{\varphi-4} + C_{12} g^k_{\varphi-3} + C_{11} g^k_{\varphi-2} + C_{10} \\ \quad \times g^k_{\varphi-1} + C_9 g^k_{\varphi} + C_8 g^k_{\varphi+1} + C_7 g^k_{\varphi+2} + C_6 \\ \quad \times g^k_{\varphi+3} + C_5 g^k_{\varphi+4} + C_4 g^k_{\varphi+5} + C_3 g^k_{\varphi+6} \\ \quad + C_2 g^k_{\varphi+7} + C_1 g^k_{\varphi+8}, \\ g^{k+1}_{4\varphi} = D_1 g^k_{\varphi-8} + D_2 g^k_{\varphi-7} + D_3 g^k_{\varphi-6} + D_4 g^k_{\varphi-5} \\ \quad + D_5 g^k_{\varphi-4} + D_6 g^k_{\varphi-3} + D_7 g^k_{\varphi-2} + D_8 g^k_{\varphi-1} \\ \quad + D_9 g^k_{\varphi} + D_{10} g^k_{\varphi+1} + D_{11} g^k_{\varphi+2} + D_{12} g^k_{\varphi+3} \\ \quad + D_{13} g^k_{\varphi+4} + D_{14} g^k_{\varphi+5} + D_{15} g^k_{\varphi+6} + D_{16} \\ \quad \times g^k_{\varphi+7} + D_{17} g^k_{\varphi+8} + D_{18} g^k_{\varphi+9}, \\ g^{k+1}_{4\varphi+1} = D_{18} g^k_{\varphi-8} + D_{17} g^k_{\varphi-7} + D_{16} g^k_{\varphi-6} + D_{15} \\ \quad \times g^k_{\varphi-5} + D_{14} g^k_{\varphi-4} + D_{13} g^k_{\varphi-3} + D_{12} g^k_{\varphi-2} \\ \quad + D_{11} g^k_{\varphi-1} + D_{10} g^k_{\varphi} + D_9 g^k_{\varphi+1} + D_8 g^k_{\varphi+2} \\ \quad + D_7 g^k_{\varphi+3} + D_6 g^k_{\varphi+4} + D_5 g^k_{\varphi+5} + D_4 g^k_{\varphi+6} \\ \quad + D_3 g^k_{\varphi+7} + D_2 g^k_{\varphi+8} + D_1 g^k_{\varphi+9}, \end{cases} \tag{26}$$

where

$$\begin{cases} C_1 = \frac{170185003}{14301288703106066939904 0000}, \\ C_2 = \frac{11618623511827}{2275205020948692467712000}, \\ C_3 = \frac{2639470801758827761}{8759539330652466000691 2000}, \\ C_4 = \frac{2685095190269185327 93}{2502725523043561714483 2000}, \\ C_5 = \frac{16029066874929664797821}{23358771548406576001843200}, \\ C_6 = \frac{11208054761546257498631183}{87595393306524660006912 0000}, \\ C_7 = \frac{3220704832824422978331 7}{3619644351509283471360 00}, \\ C_8 = \frac{418702437052457420841520 1}{159264351466408472739840 00}, \\ C_9 = \frac{678810440255672301860419}{1930476987471617851392000}, \end{cases} \tag{27}$$

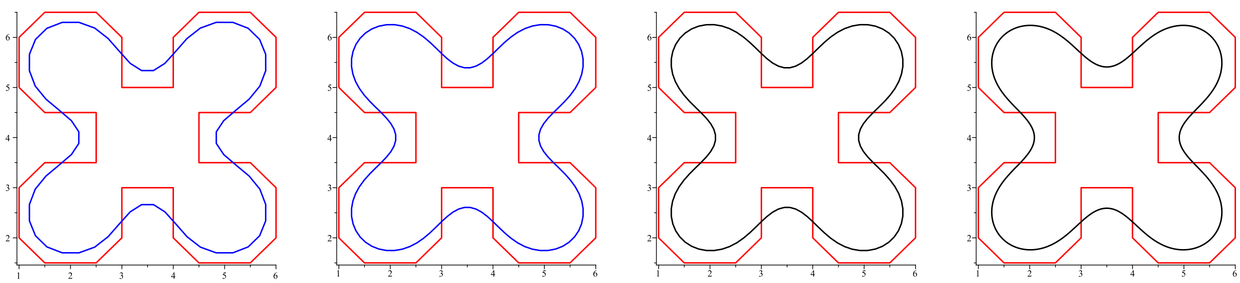
$$\begin{cases} C_{10} = \frac{68907349026172321666525 1}{31852870293281694547968 00}, \\ C_{11} = \frac{33896056277228378200493 3}{56880125523717311692800 00}, \\ C_{12} = \frac{11916341311937597342197 57}{17519078661304932001382 4000}, \\ C_{13} = \frac{87929592971882373284 69}{31852870293281694547968 000}, \\ C_{14} = \frac{1934624968539609320 3}{64885476523351600005120 00}, \\ C_{15} = \frac{1048707451741153}{2189884832663116500172 80}, \\ C_{16} = \frac{245027153519101}{87595393306524660006912 0000}, \\ C_{17} = \frac{2450263}{14015262929043945601105 92000}, \end{cases} \tag{28}$$

$$\begin{cases} D_1 = \frac{27}{4272294750508747325440000}, \\ D_2 = \frac{358812277001921}{28030525858087891202211840000}, \\ D_3 = \frac{2203625183357673913}{3503815732260986400276480000}, \\ D_4 = \frac{1034114703390726183 07}{14015262929043945601105 9200}, \\ D_5 = \frac{6376156390851085580 09}{6229005746241753600491520}, \\ D_6 = \frac{23767610644231508920682471}{70076314645219728005529 60000}, \\ D_7 = \frac{1323946982406928777310800 79}{3503815732260986400276480000}, \\ D_8 = \frac{77053158476780863751142 73}{45504100418973849354240000}, \\ D_9 = \frac{1707502598283701854724381 9}{5096459246925071127674880 0}, \end{cases} \tag{29}$$



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 2. Curves generated by the binary and quaternary subdivision schemes (19) and (21) respectively.



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 3. Curves generated by the binary and quaternary subdivision schemes (24) and (26) respectively.

$$\begin{cases} D_{10} = \dfrac{103328015159760524163023 9}{339763949795004741844992 0}, \\ D_{11} = \dfrac{40126522029222497908818157}{318528702932816945479680000}, \\ D_{12} = \dfrac{79334867894972096989478381}{3503815732260986400276480000}, \\ D_{13} = \dfrac{110789927167192437135 20413}{7007631464521972800552960000}, \\ D_{14} = \dfrac{3248016674151168146933}{934350861936263040073728 00}, \\ D_{15} = \dfrac{4496423852919659533}{28030525858087891202211840}, \\ D_{16} = \dfrac{231433236694503691}{3503815732260986400276480000}, \\ D_{17} = \dfrac{8680597683899}{2803052585808789120221184 0000}, \\ D_{18} = \dfrac{1}{2803052585808789120221184 0000}. \end{cases} \tag{30}$$

Figure 3 illustrates that the 17-point relaxed quaternary subdivision scheme (26) uses only one subdivision iterations to smooth the model while the 12-point binary subdivision scheme (24) uses two subdivision iterations to acheives that level of smoothness.

Now we present the applications of the Theorem 5 in the next few results.

Corollary 10: Here we use the result which we get from Theorem 5 for $m = 0$, Thererfore we expand the $(4m + 2)$-point binary subdivision scheme (14) for $m = 0$. Hence we get

$$\begin{cases} g_{2\varphi-1}^{k+1} = \beta_0 \; g_{\varphi-1}^{k} + \beta_2 \; g_{\varphi}^{k}, \\ g_{2\varphi}^{k+1} = \beta_2 \; g_{\varphi}^{k} + \beta_0 \; g_{\varphi+1}^{k}. \end{cases} \tag{31}$$

For the values of β_0 and β_2, we compare the 2-point binary subdivision scheme (31) with the well-known Chaikin's subdivision scheme, so we get:

$$\beta_0 = \frac{1}{4}, \quad \beta_2 = \frac{3}{4}. \tag{32}$$

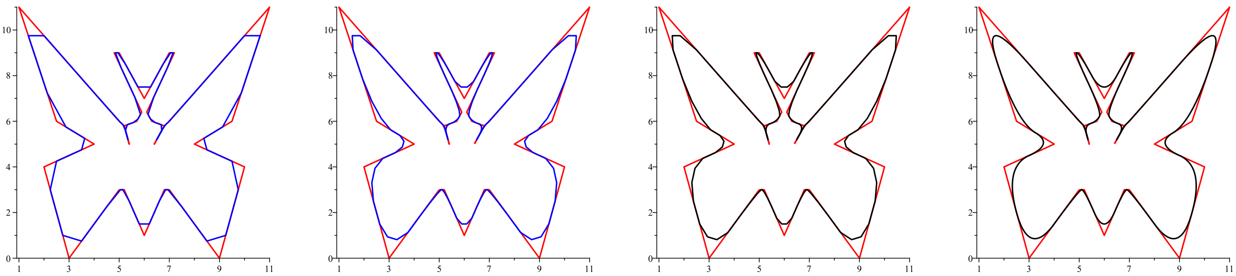
We get the 2-point relaxed quaternary subdivision scheme by putting $m = 0$ in (13) whose mask elements attained by the mask (32) of the scheme (31). The scheme is:

$$\begin{cases} g_{4\varphi-2}^{k+1} = \hat{a}_1 g_{\varphi-1}^{k} + \hat{a}_2 g_{\varphi}^{k} + \hat{a}_3 g_{\varphi+1}^{k}, \\ g_{4\varphi-1}^{k+1} = \hat{a}_3 g_{\varphi-1}^{k} + \hat{a}_2 g_{\varphi}^{k} + \hat{a}_1 g_{\varphi+1}^{k}, \\ g_{4\varphi}^{k+1} = \hat{b}_1 g_{\varphi}^{k} + \hat{b}_2 g_{\varphi+1}^{k}, \\ g_{4\varphi+1}^{k+1} = \hat{b}_2 g_{\varphi}^{k} + \hat{b}_1 g_{\varphi+1}^{k}, \end{cases} \tag{33}$$

where

$$\hat{a}_1 = \frac{3}{16}, \quad \hat{a}_2 = \frac{3}{4}, \quad \hat{a}_3 = \frac{1}{16} \quad \hat{b}_1 = \frac{5}{8}, \quad \hat{b}_2 = \frac{3}{8}.$$

Figure 4 shows the two dimensional shapes fitted by the 2-point binary subdivision scheme (31) and the 2-point relaxed quaternary subdivision scheme (33). This figure shows the speedy convergence of the quaternary subdivision scheme as compare to the binary subdivision scheme.



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 4. Curves generated by the binary and quaternary subdivision schemes (31) and (33) respectively.

Corollary 11: Let $g_\varphi^k : \varphi \in \mathbb{Z}$ be the control points at the k-th subdivision step and $g_\varphi^{k+1} : \varphi \in \mathbb{Z}$ be the refined data/points at the $(k+1)$-th refinement step. If we use $m = 1$ in the $(2m+2)$-point binary subdivision scheme (14), we get the generalized form of the following 6-point binary subdivision scheme:

$$\begin{cases} g_{2\varphi-1}^{k+1} = \beta_{-4}\ g_{\varphi-3}^k + \beta_{-2}\ g_{\varphi-2}^k + \beta_0\ g_{\varphi-1}^k \\ \quad +\beta_2\ g_\varphi^k + \beta_4\ g_{\varphi+1}^k + \beta_6\ g_{\varphi+2}^k, \\ g_{2\varphi}^{k+1} = \beta_6\ g_{\varphi-2}^k + \beta_4\ g_{\varphi-1}^k + \beta_2\ g_\varphi^k \\ \quad +\beta_0\ g_{\varphi+1}^k + \beta_{-2}\ g_{\varphi+2}^k + \beta_{-4}\ g_{\varphi+3}^k. \end{cases} \tag{34}$$

After comparing 6-point binary subdivision scheme (34) with 6-point approximating subdivision scheme by [30], we get the values of following 6 unknowns:

$$\begin{cases} \beta_{-4} = \dfrac{1}{122880}, \quad \beta_{-2} = \dfrac{3119}{122880}, \quad \beta_0 = \dfrac{6719}{20480}, \\ \beta_2 = \dfrac{31927}{61440}, \quad \beta_4 = \dfrac{15349}{122880}, \quad \beta_6 = \dfrac{81}{40960}. \end{cases} \tag{35}$$

Now if $m = 1$, the coefficients of the control points in the 8-point relaxed quaternary subdivision scheme (13) are:

$$\begin{aligned} & a_1 = \beta_6\beta_{-4}, \quad \breve{a}_2 = \beta_6\beta_{-2} + \beta_2\beta_{-4} + \beta_4\beta_6, \\ & \breve{a}_3 = \beta_6\beta_0 + \beta_2\beta_{-2} + \beta_{-2}\beta_{-4} + \beta_4^2 + \beta_0\beta_6, \\ & \breve{a}_4 = \beta_6\beta_2 + \beta_2\beta_0 + \beta_{-2}^2 + \beta_4\beta_2 + \beta_0\beta_4 \\ & \quad + \beta_{-4}\beta_6, \quad \breve{a}_5 = \beta_6\beta_4 + \beta_2^2 + \beta_{-2}\beta_0 + \beta_4\beta_0 \\ & \quad + \beta_0\beta_2 + \beta_{-4}\beta_4, \quad \breve{a}_6 = \beta_6^2 + \beta_2\beta_4 + \beta_{-2}\beta_2 \\ & \quad + \beta_4\beta_{-2} + \beta_0^2 + \beta_{-4}\beta_2, \quad \breve{a}_7 = \beta_2\beta_6 + \beta_{-2}\beta_4 \\ & \quad + \beta_4\beta_{-4} + \beta_0\beta_{-2} + \beta_{-4}\beta_0, \quad \breve{a}_8 = \beta_{-2}\beta_6 \\ & \quad + \beta_0\beta_{-4} + \beta_{-4}\beta_{-2}, \quad \breve{a}_9 = \beta_{-4}^2 \quad \breve{b}_1 = \beta_6^2 + \beta_4 \\ & \quad \times \beta_{-4}, \quad \breve{b}_2 = \beta_6\beta_4 + \beta_2\beta_6 + \beta_4\beta_{-2} + \beta_0\beta_{-4}, \\ & \breve{b}_3 = \beta_6\beta_2 + \beta_2\beta_4 + \beta_{-2}\beta_6 + \beta_4\beta_0 + \beta_0\beta_{-2} \\ & \quad + \beta_{-4}^2, \quad \breve{b}_4 = \beta_6\beta_0 + \beta_2^2 + \beta_{-2}\beta_4 + \beta_4\beta_2 + \beta_0^2 \\ & \quad + \beta_{-4}\beta_{-2}, \quad \breve{b}_5 = \beta_6\beta_{-2} + \beta_2\beta_0 + \beta_{-2}\beta_2 + \beta_4^2 \\ & \quad + \beta_0\beta_2 + \beta_{-4}\beta_0, \quad \breve{b}_6 = \beta_6\beta_{-4} + \beta_2\beta_{-2} + \beta_{-2}\beta_0 \\ & \quad + \beta_4\beta_6 + \beta_0\beta_4 + \beta_{-4}\beta_2, \quad \breve{b}_7 = \beta_2\beta_{-4} + \beta_{-2}^2 \\ & \quad + \beta_0\beta_6 + \beta_{-4}\beta_4, \quad \breve{b}_8 = \beta_{-2}\beta_{-4} + \beta_{-4}\beta_6. \end{aligned} \tag{36}$$

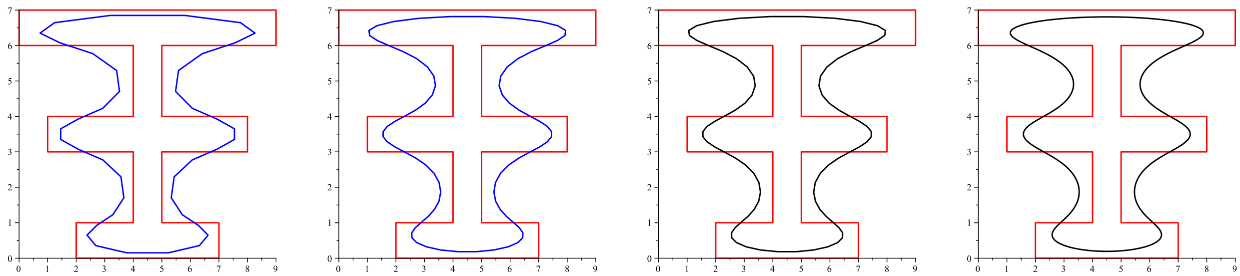
By using the values of $\beta_{-4}, \beta_{-2}, \beta_0, \beta_2, \beta_4, \beta_6$ from (35) into the (36) and after simplification, we get the following quaternary approximating subdivision scheme:

$$\begin{cases} g_{4\varphi-2}^{k+1} = \breve{a}_1 g_{\varphi-4}^k + \breve{a}_2 g_{\varphi-3}^k + \breve{a}_3 g_{\varphi-2}^k + \breve{a}_4 g_{\varphi-1}^k \\ \quad +\breve{a}_5 g_\varphi^k + \breve{a}_6 g_{\varphi+1}^k + \breve{a}_7 g_{\varphi+2}^k + \breve{a}_8 g_{\varphi+3}^k \\ \quad +\breve{a}_9 g_{\varphi+4}^k, \\ g_{4\varphi-1}^{k+1} = \breve{a}_9 g_{\varphi-4}^k + \breve{a}_8 g_{\varphi-3}^k + \breve{a}_7 g_{\varphi-2}^k + \breve{a}_6 g_{\varphi-1}^k \\ \quad +\breve{a}_5 g_\varphi^k + \breve{a}_4 g_{\varphi+1}^k + \breve{a}_3 g_{\varphi+2}^k + \breve{a}_2 g_{\varphi+3}^k \\ \quad +\breve{a}_1 g_{\varphi+4}^k, \\ g_{4\varphi}^{k+1} = \breve{b}_1 g_{\varphi-3}^k + \breve{b}_2 g_{\varphi-2}^k + \breve{b}_3 g_{\varphi-1}^k + \breve{b}_4 g_\varphi^k \\ \quad +\breve{b}_5 g_{\varphi+1}^k + \breve{b}_6 g_{\varphi+2}^k + \breve{b}_7 g_{\varphi+3}^k + \breve{b}_8 g_{\varphi+4}^k, \\ g_{4\varphi+1}^{k+1} = \breve{b}_8 g_{\varphi-3}^k + \breve{b}_7 g_{\varphi-2}^k + \breve{b}_6 g_{\varphi-1}^k + \breve{b}_5 g_\varphi^k \\ \quad +\breve{b}_4 g_{\varphi+1}^k + \breve{b}_3 g_{\varphi+2}^k + \breve{b}_2 g_{\varphi+3}^k + \breve{b}_1 g_{\varphi+4}^k, \end{cases} \tag{37}$$

where

$$\begin{cases} \breve{a}_1 = \dfrac{27}{1677721600}, \ \breve{a}_2 = \dfrac{2275789}{7549747200}, \ \breve{a}_3 = \dfrac{9086963}{301989888}, \\ \breve{a}_4 = \dfrac{699721619}{2516582400}, \ \breve{a}_5 = \dfrac{369990379}{754974720}, \ \breve{a}_6 = \dfrac{1426235351}{7549747200}, \\ \breve{a}_7 = \dfrac{31530847}{2516582400}, \ \breve{a}_8 = \dfrac{16027}{301989888}, \ \breve{a}_9 = \dfrac{1}{15099494400} \\ \breve{b}_1 = \dfrac{37199}{7549747200}, \ \breve{b}_2 = \dfrac{33580087}{7549747200}, \ \breve{b}_3 = \dfrac{290148073}{2516582400}, \\ \breve{b}_4 = \dfrac{674031991}{1509949440}, \ \breve{b}_5 = \dfrac{558397097}{1509949440}, \ \breve{b}_6 = \dfrac{157912247}{2516582400}, \\ \breve{b}_7 = \dfrac{9801833}{7549747200}, \ \breve{b}_8 = \dfrac{1681}{7549747200}. \end{cases} \tag{38}$$

The graphical results of the 6-point binary subdivision schemes (34) and the 8-point relaxed quaternary approximating subdivision schemes (37) are reviewed in Figure 5. It is clear that 8-point relaxed quaternary approximating subdivision scheme uses less iterations for smoothness



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 5. Curves generated by the binary and quaternary subdivision schemes (34) and (37) respectively.

comparatively to its corresponding 6-point binary subdivision scheme.

Corollary 12: The binary subdivision scheme (14) for $m = 2$ reduces to an approximating subdivsion scheme whose each subdivision rule use the linear combination of 10 control points of subdivision level k to get a control point at subdivision level $k+1$. So the general form of this scheme is:

$$\left\{\begin{aligned} g^{k+1}_{2\varphi-1} &= \beta_{-8}\ g^k_{\varphi-5} + \beta_{-6}\ g^k_{\varphi-4} + \beta_{-4}\ g^k_{\varphi-3} + \beta_{-2} \\ &\quad \times g^k_{\varphi-2} + \beta_0\ g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_4\ g^k_{\varphi+1} + \\ &\quad + \beta_6\ g^k_{\varphi+2} + \beta_8\ g^k_{\varphi+3} + \beta_{10}\ g^k_{\varphi+4}, \\ g^{k+1}_{2\varphi} &= \beta_{10}\ g^k_{\varphi-4} + \beta_8\ g^k_{\varphi-3} + \beta_6\ g^k_{\varphi-2} + \beta_4 \\ &\quad \times g^k_{\varphi-1} + \beta_2\ g^k_{\varphi} + \beta_0\ g^k_{\varphi+1} + \beta_{-2}\ g^k_{\varphi+2} \\ &\quad + \beta_{-4}\ g^k_{\varphi+3} + \beta_{-6}\ g^k_{\varphi+4} + \beta_{-8}\ g^k_{\varphi+5}. \end{aligned}\right. \tag{39}$$

To get the values of 10 unknowns $\beta_{-8}, \beta_{-6} \dots \beta_8,\ \beta_{10}$, we uses the 10-point binary subdivision scheme presented by [30] so that the control points at level $k+1$ in (39) become the convex combination of the control points of level k. Hence the values of unknowns are:

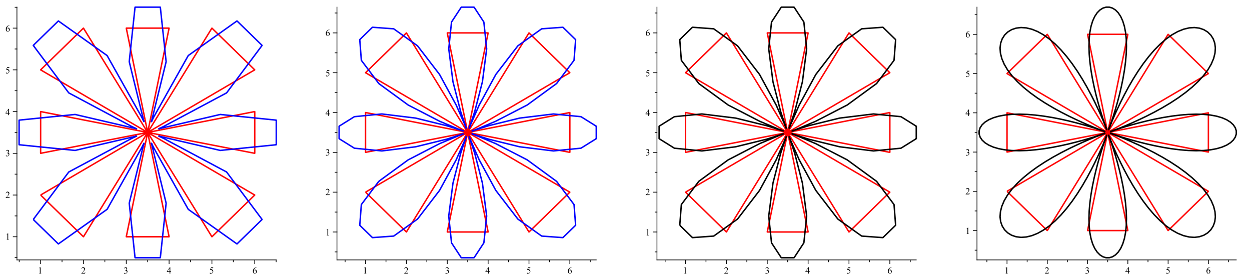
$$\left\{\begin{aligned} &\beta_{-8} = \frac{12155}{33554432},\ \beta_{-6} = \frac{-138567}{33554432},\ \beta_{-4} = \frac{188955}{8388608}, \\ &\beta_{-2} = \frac{-692835}{8388608},\ \beta_0 = \frac{4849845}{16777216},\ \beta_2 = \frac{14549535}{16777216}, \\ &\beta_4 = \frac{-969969}{8388608},\ \beta_6 = \frac{230945}{8388608},\ \beta_8 = \frac{-159885}{33554432}, \\ &\beta_{10} = \frac{13585}{33554432}. \end{aligned}\right. \tag{40}$$

Now we simplify (13) for $m = 2$ and get the general form of the 14-point relaxed quaternary subdivision scheme. Now we use the unknowns from (40) into (13) and get the following quaternary subdivision scheme which is relaxed bacasue its two subdivision rules are the convex combination of 14 control points of level k whereas the remaining two subdivision rules are the convex combination of 15 control points of level k.

$$\left\{\begin{aligned} g^{k+1}_{4\varphi-2} &= \tilde{a}_1 g^k_{\varphi-7} + \tilde{a}_2 g^k_{\varphi-6} + \tilde{a}_3 g^k_{\varphi-5} + \tilde{a}_4 g^k_{\varphi-4} \\ &\quad + \tilde{a}_5 g^k_{\varphi-3} + \tilde{a}_6 g^k_{\varphi-2} + \tilde{a}_7 g^k_{\varphi-1} + \tilde{a}_8 g^k_{\varphi} \\ &\quad + \tilde{a}_9 g^k_{\varphi+1} + \tilde{a}_{10} g^k_{\varphi+2} + \tilde{a}_{11} g^k_{\varphi+3} + \tilde{a}_{12} \\ &\quad \times g^k_{\varphi+4} + \tilde{a}_{13} g^k_{\varphi+5} + \tilde{a}_{14} g^k_{\varphi+6} + \tilde{a}_{15} \\ &\quad \times g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi-1} &= \tilde{a}_{15} g^k_{\varphi-7} + \tilde{a}_{14} g^k_{\varphi-6} + \tilde{a}_{13} g^k_{\varphi-5} + \tilde{a}_{12} \\ &\quad \times g^k_{\varphi-4} + \tilde{a}_{11} g^k_{\varphi-3} + \tilde{a}_{10} g^k_{\varphi-2} + \tilde{a}_9 g^k_{\varphi-1} \\ &\quad + \tilde{a}_8 g^k_{\varphi} + \tilde{a}_7 g^k_{\varphi+1} + \tilde{a}_6 g^k_{\varphi+2} + \tilde{a}_5 g^k_{\varphi+3} \\ &\quad + \tilde{a}_4 g^k_{\varphi+4} + \tilde{a}_3 g^k_{\varphi+5} + \tilde{a}_2 g^k_{\varphi+6} + \tilde{a}_1 g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi} &= \tilde{b}_1 g^k_{\varphi-6} + \tilde{b}_2 g^k_{\varphi-5} + \tilde{b}_3 g^k_{\varphi-4} + \tilde{b}_4 g^k_{\varphi-3} \\ &\quad + \tilde{b}_5 g^k_{\varphi-2} + \tilde{b}_6 g^k_{\varphi-1} + \tilde{b}_7 g^k_{\varphi} + \tilde{b}_8 g^k_{\varphi+1} \\ &\quad + \tilde{b}_9 g^k_{\varphi+2} + \tilde{b}_{10} g^k_{\varphi+3} + \tilde{b}_{11} g^k_{\varphi+4} + \tilde{b}_{12} \\ &\quad \times g^k_{\varphi+5} + \tilde{b}_{13} g^k_{\varphi+6} + \tilde{b}_{14} g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi+1} &= \tilde{b}_{14} g^k_{\varphi-6} + \tilde{b}_{13} g^k_{\varphi-5} + \tilde{b}_{12} g^k_{\varphi-4} + \tilde{b}_{11} \\ &\quad \times g^k_{\varphi-3} + \tilde{b}_{10} g^k_{\varphi-2} + \tilde{b}_9 g^k_{\varphi-1} + \tilde{b}_8 g^k_{\varphi} \\ &\quad + \tilde{b}_7 g^k_{\varphi+1} + \tilde{b}_6 g^k_{\varphi+2} + \tilde{b}_5 g^k_{\varphi+3} + \tilde{b}_4 g^k_{\varphi+4} \\ &\quad + \tilde{b}_3 g^k_{\varphi+5} + \tilde{b}_2 g^k_{\varphi+6} + \tilde{b}_1 g^k_{\varphi+7}, \end{aligned}\right. \tag{41}$$

where

$$\left\{\begin{aligned} &\tilde{a}_1 = \frac{165125675}{1125899906842624},\quad \tilde{a}_2 = \frac{896759435}{140737488355328}, \\ &\tilde{a}_3 = \frac{208816685055}{1125899906842624},\quad \tilde{a}_4 = \frac{-350111003385}{140737488355328}, \\ &\tilde{a}_5 = \frac{15443267900775}{1125899906842624},\quad \tilde{a}_6 = \frac{-845664470365}{17592186044416}, \\ &\tilde{a}_7 = \frac{165267038051115}{1125899906842624},\quad \tilde{a}_8 = \frac{66176709385215}{70368744177664}, \\ &\tilde{a}_9 = \frac{-67374962898815}{1125899906842624},\quad \tilde{a}_{10} = \frac{1418419563275}{140737488355328}, \\ &\tilde{a}_{11} = \frac{-772276338691}{1125899906842624},\quad \tilde{a}_{12} = \frac{-17904682965}{140737488355328}, \\ &\tilde{a}_{13} = \frac{6861145005}{1125899906842624},\quad \tilde{a}_{14} = \frac{351267345}{70368744177664}, \end{aligned}\right. \tag{42}$$



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 6. Curves generated by the binary and quaternary subdivision schemes (39) and (41) respectively.

$$\begin{cases} \tilde{a}_{15} = \dfrac{147744025}{1125899906842624}\ \tilde{b}_1 = \dfrac{-879424975}{562949953421312}, \\ \tilde{b}_2 = \dfrac{-7313797205}{562949953421312},\ \tilde{b}_3 = \dfrac{198710020223}{281474976710656}, \\ \tilde{b}_4 = \dfrac{-1928438555615}{281474976710656},\ \tilde{b}_5 = \dfrac{20280883755275}{562949953421312}, \\ \tilde{b}_6 = \dfrac{-78511743180975}{562949953421312},\ \tilde{b}_7 = \dfrac{105752643189765}{140737488355328}, \\ \tilde{b}_8 = \dfrac{63006444414771}{140737488355328},\ \tilde{b}_9 = \dfrac{-64970510830665}{562949953421312}, \\ \tilde{b}_{10} = \dfrac{17600920061725}{562949953421312},\ \tilde{b}_{11} = \dfrac{-1671012940025}{281474976710656}, \\ \tilde{b}_{12} = \dfrac{163784577385}{281474976710656},\ \tilde{b}_{13} = \dfrac{-3080205843}{562949953421312}, \\ \tilde{b}_{14} = \dfrac{-759578105}{562949953421312}. \end{cases} \tag{43}$$

In Figure 6, we present the graphical shapes generated by the binary subdivision scheme (39) and the quaternary approximating subdivision scheme (41). The difference at first two subdivision levels can be visualized clearly from this figure.

Corollary 13: This corollary also shows the application of Theorem 5 for $m = 2$. So the general form of the 10-point binary subdivision scheme is same as given in (39) of Corollary 12 and the values of unknowns can be get by [30], that are:

$$\begin{cases} \beta_{-8} = \dfrac{1}{95126814720},\ \beta_{-6} = \dfrac{390623}{19025362944}, \\ \beta_{-4} = \dfrac{13138903}{3397386240},\ \beta_{-2} = \dfrac{1704546247}{23781703680}, \\ \beta_0 = \dfrac{14871214991}{47563407360},\ \beta_2 = \dfrac{19761725357}{47563407360}, \\ \beta_4 = \dfrac{833871641}{4756340736},\ \beta_6 = \dfrac{488824339}{23781703680}, \\ \beta_8 = \dfrac{40156777}{95126814720},\ \beta_{10} = \dfrac{243}{1174405120}. \end{cases} \tag{44}$$

Hence the coefficients of the following 14-point relaxed quaternary subdivision scheme which we get from (13) for $m = 2$ are the non-linear combination of the values given in (44)

$$\begin{cases} g^{k+1}_{4\varphi-2} &= c_1 g^k_{\varphi-7} + c_2 g^k_{\varphi-6} + c_3 g^k_{\varphi-5} + c_4 g^k_{\varphi-4} \\ &\quad + c_5 g^k_{\varphi-3} + c_6 g^k_{\varphi-2} + c_7 g^k_{\varphi-1} + c_8 g^k_{\varphi} + c_9 \\ &\quad \times g^k_{\varphi+1} + c_{10} g^k_{\varphi+2} + c_{11} g^k_{\varphi+3} + c_{12} g^k_{\varphi+4} \\ &\quad + c_{13} g^k_{\varphi+5} + c_{14} g^k_{\varphi+6} + c_{15} g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi-1} &= c_{15} g^k_{\varphi-7} + c_{14} g^k_{\varphi-6} + c_{13} g^k_{\varphi-5} + c_{12} \\ &\quad \times g^k_{\varphi-4} + c_{11} g^k_{\varphi-3} + c_{10} g^k_{\varphi-2} + c_9 g^k_{\varphi-1} \\ &\quad + c_8 g^k_{\varphi} + c_7 g^k_{\varphi+1} + c_6 g^k_{\varphi+2} + c_5 g^k_{\varphi+3} + c_4 \\ &\quad \times g^k_{\varphi+4} + c_3 g^k_{\varphi+5} + c_2 g^k_{\varphi+6} + c_1 g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi} &= d_1 g^k_{\varphi-6} + d_2 g^k_{\varphi-5} + d_3 g^k_{\varphi-4} + d_4 g^k_{\varphi-3} \\ &\quad + d_5 g^k_{\varphi-2} + d_6 g^k_{\varphi-1} + d_7 g^k_{\varphi} + d_8 g^k_{\varphi+1} + d_9 \\ &\quad \times g^k_{\varphi+2} + d_{10} g^k_{\varphi+3} + d_{11} g^k_{\varphi+4} + d_{12} g_{\varphi+5} \\ &\quad + d_{13} g_{\varphi+6} + d_{14} g^k_{\varphi+7}, \\ g^{k+1}_{4\varphi+1} &= d_{14} g^k_{\varphi-6} + d_{13} g^k_{\varphi-5} + d_{12} g^k_{\varphi-4} + d_{11} g^k_{\varphi-3} \\ &\quad + d_{10} g^k_{\varphi-2} + d_9 g^k_{\varphi-1} + d_8 g^k_{\varphi} + d_7 g^k_{\varphi+1} \\ &\quad + d_6 g^k_{\varphi+2} + d_5 g^k_{\varphi+3} + d_4 g^k_{\varphi+4} + d_3 g^k_{\varphi+5} \\ &\quad + d_2 g^k_{\varphi+6} + d_1 g^k_{\varphi+7}, \end{cases} \tag{45}$$

where

$$\begin{cases} c_1 = \dfrac{3}{1379227385882214400}, \\ c_2 = \dfrac{103850537699}{1131138859846651084800}, \\ c_3 = \dfrac{384468662194361}{603274058584880578560}, \\ c_4 = \dfrac{193184601091081183}{1131138859846651084800}, \\ c_5 = \dfrac{623591548016510769 91}{9049110878773208678400}, \\ c_6 = \dfrac{1043807155937640 0791}{1413923574808313856 00}, \\ c_7 = \dfrac{1169706201242893959 91}{4309100418463432704 00}, \\ c_8 = \dfrac{137915687699480465}{359091701538619392}, \end{cases} \tag{46}$$

$$\begin{cases} c_9 = \dfrac{6484620539770876423339}{30163702929244028928000}, \\ c_{10} = \dfrac{171255196344729908817}{377046286615550361600}, \\ c_{11} = \dfrac{28434156344565775493}{904911087877320867840 0}, \\ c_{12} = \dfrac{19659422070653153}{377046286615550361600}, \\ c_{13} = \dfrac{186238665573617}{180982217575464173568 0}, \\ c_{14} = \dfrac{346544687}{807956328461893632 00}, \\ c_{15} = \dfrac{1}{904911087877320867840 0}, \end{cases} \tag{47}$$

$$\begin{cases} d_1 = \dfrac{213788633}{452455543938660433920 0}, \\ d_2 = \dfrac{8408857474501}{646365062769514905600}, \\ d_3 = \dfrac{646332081504007}{461689330549653504 00}, \\ d_4 = \dfrac{989843345911370179}{754092573231100723200}, \\ d_5 = \dfrac{236010835312030392 71}{904911087877320867840}, \\ d_6 = \dfrac{24171688002822559624 3}{150818514646220144640 0}, \\ d_7 = \dfrac{136704671280724852463}{377046286615550361600}, \end{cases} \tag{48}$$

$$\begin{cases} d_8 = \dfrac{405978679693950532 03}{125682095538516787200}, \\ d_9 = \dfrac{241962547518165047 87}{215455020923171635200}, \\ d_{10} = \dfrac{180071808751776040 7}{129273012553902981120}, \\ d_{11} = \dfrac{112821727257578191 9}{226227771969330216960 0}, \\ d_{12} = \dfrac{7348484779529681}{226227771969330216960 0}, \\ d_{13} = \dfrac{1843780220327}{150818514646220144640 0}, \\ d_{14} = \dfrac{986399}{452455543938660433920 0}. \end{cases} \tag{49}$$

Figure 7 also give the comparison between the models generated by the 10-point binary and the 14-point relaxed quaternary subdivision schemes when masks of the subdivision schemes are all positive.

IV. HÖLDER'S REGULARITY OF THE PRESENTED PAIRS OF SUBDIVISION SCHEMES

In this section, we evaluate and compare the Hölder's regularity of the each pair of binary and quaternary subdivision schemes which we have discussed in the corollaries of Section III. This evaluation is done by a well-known technique presented by [21] which is defined here.

Definition 14: Hölder regularity is an extension of the Laurent's polynomial of continuity which gives more information about any scheme. The Hölder regularity of subdivision scheme with Laurent's polynomial $\mu(c)$ can be computed in the following way. Let

$$\mu(c) = \left(\frac{1+c+c^2+\ldots+c^{s-1}}{s}\right)^p \nu(c),$$

without loss of generality, we can suppose that e_0, e_1,..., e_{t-1},e_t to be the non-zero coefficients of $\nu(c)$ and let E_0, E_1, ..., E_{t-1}, E_t be the $t \times t$ matrices with elements:

$$(E_q)_{ij} = e_{t+i-sj+q}, \;\; i,j = 1,2,\ldots,t \;\; and \;\; q = 0,1,\ldots,t. \tag{50}$$

Then the Hölder regularity of the subdivision schemes is given by

$$r = p - \log_s(\xi).$$

where ξ is the joint spectral radius of the matrices E_0, E_1, ..., E_{t-1}, E_t, i.e

$$\begin{aligned} \xi &= \rho(E_0, E_1, \ldots, E_{t-1}, E_t) \\ &= \lim_{l\to\infty} \sup(\max\{\|E_{i(l)}E_{i(l-1)}\ldots E_{i(2)}E_{i(1)}\|_\infty^{\frac{1}{l}} : \; i_l \in [0,1]\}). \end{aligned}$$

and

$$\max\{\rho(E_0),\ldots,\rho(E_t)\} \le \max\{\|E_0\|_\infty,\ldots,\|E_t\|_\infty\} \tag{51}$$

Since ξ is bounded from below by the spectral radii and above from the norm of the matrices E_0, E_1, ..., E_{t-1}, E_t.

Then

$$\max\{\rho(E_0),\ldots,\rho(E_t)\} \le \xi \le \max\{\|E_0\|_\infty,\ldots,\|E_t\|_\infty\}.$$

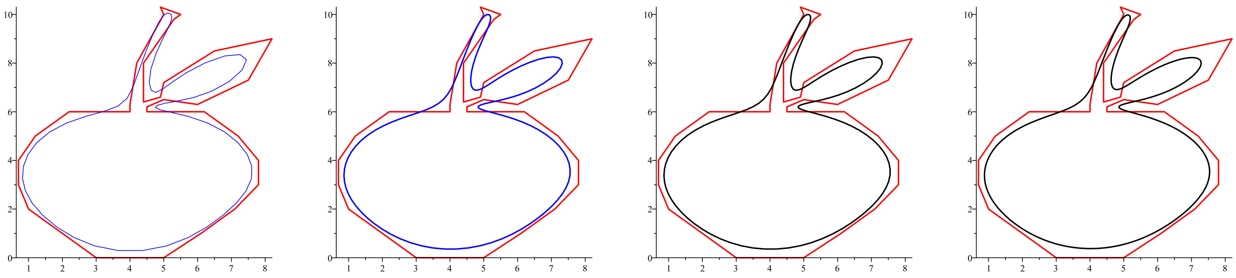
Given is an important remark about the Laurent polynomial representation of the binary and quaternary subdivision schemes.

Remark 15: Througout the paper, the Laurent polynomial of the $4m$-point binary subdivision scheme is denoted by $\mu_{4m}(c)$, while the Laurent polynomial of the $(4m+2)$-point binary subdivision scheme is denoted by $\mu_{4m+2}(c)$. In the same way, the Laurent Polynomial of the $(6m-1)$-point relaxed quaternary schemes is denoted by $U_{6m-1}(c)$, while the Laurent polynomial of the $(6m+2)$-point relaxed quaternary scheme is denoted by the symbol $U_{6m+2}(c)$.

In the following theorem, we estimate the Hölder's continuity of the 4-point binary and its corresponding 5-point relaxed quaternary subdivision schemes.

Theorem 16: The Hölder's regularity of the 4-point binary subdivision scheme (15) is 4.124809715, whereas the Hölder's regularity of the 5-point relaxed quaternary subdivision scheme (17) is 4.12397897.

Proof: To follow the procedure for Hölder's regularity, firstly we write the Laurent's polynomial $\mu_4(c)$ of the binary



(a) One SS by BSS (b) One SS by QSS (c) Two SSs by BSS (d) Two SSs by QSS

FIGURE 7. Curves generated by the pair of subdivision schemes given in Corollary 13.

subdivision scheme (15). That is:

$$\mu_4(c) = \frac{1}{384}(c^{-3} + 27c^{-2} + 121c^{-1} + 235c^0 + 235c^1 + 121c^2 + 27c^3 + c^4).$$

This implies that

$$\mu_4(c) = \left(\frac{1+c}{2}\right)^5 \nu_4(c), \tag{52}$$

where

$$\nu_4(c) = \frac{1 + 22c + c^2}{12c^3}. \tag{53}$$

Now from (50) we know that $(E_q)_{ij} = e_{t+i-sj+q}$, so by (53) we have
$e_0 = \frac{1}{12}$, $e_1 = \frac{11}{6}$, $e_2 = \frac{1}{12}$, $p = 5$, $t = 2$ and $s = 2$, thus $q = 0, 1, 2$ and then E_0, E_1 and E_2 are the matrices with the elements:

$$\begin{cases} (E_0)_{ij} = e_{2+i-2j}, \\ (E_1)_{ij} = e_{2+i-2j+1}, \\ (E_2)_{ij} = e_{2+i-2j+2}, \end{cases}$$

where $i, j = 1, 2$.

Hence

$$E_0 = \begin{bmatrix} \frac{11}{6} & 0 \\ \frac{1}{12} & \frac{1}{12} \end{bmatrix}, \quad E_1 = \begin{bmatrix} \frac{1}{12} & \frac{1}{12} \\ 0 & \frac{11}{6} \end{bmatrix} \text{ and}$$

$$E_2 = \begin{bmatrix} 0 & \frac{11}{6} \\ 0 & \frac{1}{12} \end{bmatrix}.$$

Now we calculate the largest eigenvalues of E_0, E_1 and E_2, that are:

$$\rho(E_0) = 1.8333, \quad \rho(E_1) = 1.8333, \quad \rho(E_2) = 0.0833.$$

Further, the norm-infinity of these three matrices are:

$$||E_0||_\infty = 1.8352, \quad ||E_1||_\infty = 1.8352, \quad ||E_2||_\infty = 1.8352.$$

By using (51), we have

$$\max\{1.8333, 1.8333, 0.0833\} \le \xi \le \max\{1.8352, 1.8352, 1.8352\}.$$

This implies that

$$\xi = 1.834250000$$

Thus the Hölder's regularity of scheme (15) is:

$$r = p - log_s(\xi) = 5 - log_2(1.834250000) = 4.124809715.$$

The Laurent polynomial $U_5(c)$ of the quaternary subdivision scheme (17) is

$$U_5(c) = \hat{B}_6c^{-10} + \hat{B}_1c^{-9} + \hat{A}_5c^{-8} + \hat{A}_1c^{-7} + \hat{B}_5c^{-6} + \hat{B}_2c^{-5} + \hat{A}_4c^{-4} + \hat{A}_2c^{-3} + \hat{B}_4c^{-2} + \hat{B}_3c^{-1} + \hat{A}_3c^0 + \hat{A}_3c^1 + \hat{B}_3c^2 + \hat{B}_4c^3 + \hat{A}_2c^4 + \hat{A}_4c^5 + \hat{B}_2c^6 + \hat{B}_5c^7 + \hat{A}_1c^8 + \hat{A}_5c^9 + \hat{B}_1c^{10} + \hat{B}_6c^{11},$$

where the values of $\hat{A}_1, \ldots, \hat{A}_5$, and $\hat{B}_1, \ldots, \hat{B}_6$ are given in (18). This implies that

$$U_5(c) = \left(\frac{1 + c + c^2 + c^3}{4}\right)^5 V_5(c), \tag{54}$$

where

$$V_5(c) = \frac{1}{144c^{10}}(1 + 22c + 23c^2 + 484c^3 + 23c^4 + 22c^5 + c^6).$$

It is given from (50) that $(E_q)_{ij} = e_{t+i-sj+q}$, so by (54), we have
$e_0 = \frac{1}{144}$, $e_1 = \frac{11}{72}$, $e_2 = \frac{23}{144}$, $e_3 = \frac{121}{36}$, $e_4 = \frac{23}{144}$, $e_5 = \frac{11}{72}$, $e_6 = \frac{1}{144}$, $p = 5$, $t = 6$ and $s = 4$. Thus $q = 0, 1, 2, \ldots, 6$ and then $E_0, E_1, \ldots, E_6$ are the matrices with the elements:

$$\begin{cases} (E_0)_{ij} = e_{6+i-4j} \\ (E_1)_{ij} = e_{6+i-4j+1}, \\ \vdots \\ (E_5)_{ij} = e_{6+i-4j+5}, \\ (E_6)_{ij} = e_{6+i-4j+6}, \end{cases} \tag{55}$$

where $i, j = 1, 2, 3, 4, 5, 6$.

So, we have

$$E_0 = \begin{bmatrix} e_3 & 0 & 0 & 0 & 0 & 0 \\ e_4 & e_0 & 0 & 0 & 0 & 0 \\ e_5 & e_1 & 0 & 0 & 0 & 0 \\ e_6 & e_2 & 0 & 0 & 0 & 0 \\ 0 & e_3 & 0 & 0 & 0 & 0 \\ 0 & e_4 & e_0 & 0 & 0 & 0 \end{bmatrix},$$

$$E_1 = \begin{bmatrix} e_4 & e_0 & 0 & 0 & 0 & 0 \\ e_5 & e_1 & 0 & 0 & 0 & 0 \\ e_6 & e_2 & 0 & 0 & 0 & 0 \\ 0 & e_3 & 0 & 0 & 0 & 0 \\ 0 & e_4 & e_0 & 0 & 0 & 0 \\ 0 & e_5 & e_1 & 0 & 0 & 0 \end{bmatrix},$$

$$E_2 = \begin{bmatrix} e_5 & e_1 & 0 & 0 & 0 & 0 \\ e_6 & e_2 & 0 & 0 & 0 & 0 \\ 0 & e_3 & 0 & 0 & 0 & 0 \\ 0 & e_4 & e_0 & 0 & 0 & 0 \\ 0 & e_5 & e_1 & 0 & 0 & 0 \\ 0 & e_6 & e_2 & 0 & 0 & 0 \end{bmatrix},$$

$$E_3 = \begin{bmatrix} e_6 & e_2 & 0 & 0 & 0 & 0 \\ 0 & e_3 & 0 & 0 & 0 & 0 \\ 0 & e_4 & e_0 & 0 & 0 & 0 \\ 0 & e_5 & e_1 & 0 & 0 & 0 \\ 0 & e_6 & e_2 & 0 & 0 & 0 \\ 0 & 0 & e_3 & 0 & 0 & 0 \end{bmatrix},$$

$$E_4 = \begin{bmatrix} 0 & e_3 & 0 & 0 & 0 & 0 \\ 0 & e_4 & e_0 & 0 & 0 & 0 \\ 0 & e_5 & e_1 & 0 & 0 & 0 \\ 0 & e_6 & e_2 & 0 & 0 & 0 \\ 0 & 0 & e_3 & 0 & 0 & 0 \\ 0 & 0 & e_4 & e_0 & 0 & 0 \end{bmatrix},$$

$$E_5 = \begin{bmatrix} 0 & e_4 & e_0 & 0 & 0 & 0 \\ 0 & e_5 & e_1 & 0 & 0 & 0 \\ 0 & e_6 & e_2 & 0 & 0 & 0 \\ 0 & 0 & e_3 & 0 & 0 & 0 \\ 0 & 0 & e_4 & e_0 & 0 & 0 \\ 0 & 0 & e_5 & e_1 & 0 & 0 \end{bmatrix}$$

$$\text{and } E_6 = \begin{bmatrix} 0 & e_5 & e_1 & 0 & 0 & 0 \\ 0 & e_6 & e_2 & 0 & 0 & 0 \\ 0 & 0 & e_3 & 0 & 0 & 0 \\ 0 & 0 & e_4 & e_0 & 0 & 0 \\ 0 & 0 & e_5 & e_1 & 0 & 0 \\ 0 & 0 & e_6 & e_2 & 0 & 0 \end{bmatrix}$$

The largest eignevalues of $E_0, E_1, \ldots, E_6$ are:

$$\rho(E_0) = 3.3611, \quad \rho(E_1) = 0.1890, \quad \rho(E_2) = 0.1890,$$
$$\rho(E_3) = 3.3611, \quad \rho(E_4) = 0.1890, \quad \rho(E_5) = 0.1890,$$
$$\rho(E_6) = 3.3611.$$

The norm-infinity of matrices $E_0, E_1, \ldots, E_6$ are:

$$||E_0||_\infty = 3.3745, \quad ||E_1||_\infty = 3.3756, \quad ||E_2||_\infty = 3.3756,$$
$$||E_3||_\infty = 3.3745, \quad ||E_4||_\infty = 3.3745, \quad ||E_5||_\infty = 3.3756,$$
$$||E_6||_\infty = 3.3756.$$

Now from (51), we have

$$\max[\rho(E_0), \rho(E_1), \ldots, \rho(E_5), \rho(E_6)] \leq \xi$$
$$\leq \max[||E_0||_\infty, ||E_1||_\infty, \ldots, ||E_5||_\infty, ||E_6||_\infty].$$

This implies that

$$\max[3.3611, 0.1890, \ldots, 0.1890, 3.3611] \leq \xi$$
$$\leq \max[3.3745, 3.3756, \ldots, 3.3756, 3.3756].$$

Since the largest eigenvalue and the max-norm of the matrices is between (3.3611 and 3.3756) and we choose the mid value 3.368350000 of the above given values, so the Hölder regularity is given by

$$r = p - log_s(\xi) = 5 - log_4(3.368350000) = 4.123978973.$$

□

The proof of the following theorems follows the proof of Theorem 16.

Theorem 17: The Hölder's regularity of the 6*-point binary subdivision scheme (34) is 6.383689358, while the Hölder's regularity of the* 8*-point relaxed quaternary subdivision scheme (37) is 6.378805452.*

Theorem 18: The Hölder's regularity of the 8*-point binary subdivision scheme (19) is 8.575077912 and the Hölder's regularity of the* 11*-point relaxed quaternary subdivision scheme (21) is 8.561638397.*

Theorem 19: The Hölder's regularities of the 10*-point binary subdivision scheme (39) and the* 14*-point relaxed quaternary subdivision scheme (41) are 3.768111637 and 4.571743466 respectively.*

Theorem 20: The Hölder's regularity of the 10*-point binary subdivision scheme corresponding to the mask (44) is 10.67905327, although the Hölder's regularity of the* 14*-point relaxed quaternary subdivision scheme (45) is 10.65483615.*

Theorem 21: The Hölder's regularity of the 12*-point binary subdivision scheme (24) is 12.72368201, whereas the Hölder's regularity of the* 17*-point relaxed quaternary subdivision scheme (26) is 12.69332847.*

V. DEGREE OF PRECISION

Degree of precision of a subdivision scheme is the ability of a subdivision scheme to produce the same polynomial from which the initial data is taken. In other words, degree of precision of a subdivision scheme is n if it produces polynomials of degree $0, 1, \ldots, n$ when the initial data is taken from these polynomials respectively, but not produces the polynomial of degree $n+1$ when initial data is taken from that specific polynomial of degree $n+1$. Whereas the degree of polynomial generation of a subdivision scheme is its ability to produces the polynomial of same degree from which the initial data is chosen.

In this section, we discuss the response of the pair of binary and quaternary schemes on polynomial data. We summerize these responses in Table 1. In this table, BSS, QSS, DoP and DoG denote binary subdivision scheme, quaternary

TABLE 1. Response of the pairs of binary and quaternary subdivision schemes to the polynomial data.

BSS	DoP	DoG	QSS	DoP	DoG
2-point (31)	1	2	2-point relaxed (33)	1	2
4-point (15)	1	4	5-point relaxed (17)	1	4
6-point (34)	1	6	8-point relaxed (37)	1	6
8-point (19)	1	8	11-point relaxed (21)	1	8
10-point (39)	9	10	14-point relaxed (41)	9	10
10-point (44)	1	10	14-point relaxed (45)	1	10
12-point (24)	1	12	17-point relaxed (26)	1	12

subdivision scheme, degree of precision and degree of polynomial generation respectively. The table indicates that there is no impact on these two characteristics of the pair of schemes when we move from binary to its corresponding quaternary subdivision scheme.

VI. CONCLUSION

In this research, we presented a new study about the binary and quaternary subdivision schemes. We proved that every even-point binary subdivision scheme can be used to get a relaxed quaternary subdivision scheme. We presented an intresting link between the masks of theses pairs of schemes, that the mask of the quaternary scheme is just the non-linear combination of the mask of the binary subdivision scheme. The results are applicable on all linear and stationary even-point binary subdivision schemes without any restriction. Moreover, we validated our general results by different even-point binary subdivision schemes.The graphical inspections and theoratical analysis of the pairs of schemes are also presented. Which shows that the final models of both type of schemes are almost same but quaternary subdivision schemes give us final models in less number of iterations as compare to the parent binary subdivision schemes.

A. FUTURE TRENDS, GAPS, FEASIBILITIES, AND LIMITATIONS

In looking to the future, it is essential to explore the conversion methods between different arity and complexity subdivision schemes beyond the binary to quaternary transition currently in progress. This expansion could address the gap in understanding how various subdivision schemes interrelate and can be optimized for different geometric modeling tasks. While our research has demonstrated the feasibility of converting even-point binary schemes to quaternary schemes, similar methodologies could be applied to ternary, quintary, or even higher arity schemes. However, this comes with limitations, such as increased computational complexity and the need for extensive validation across various types of geometric data. Practical implementation challenges also exist, including ensuring compatibility with existing modeling software and maintaining efficiency. Despite these hurdles, exploring these conversion methods holds significant promise for advancing geometric modeling, making it more adaptable and efficient for diverse applications.

AUTHORS' CONTRIBUTIONS

Conceptualization: Rabia Hameed; Formal analysis: Sidra Nosheen, Rabia Hameed, and Jihad Younis; Methodology: Sidra Nosheen and Rabia Hameed; Supervision: Rabia Hameed; Writing original draft, Sidra Nosheen and Rabia Hameed; Writing, review and editing: Sidra Nosheen, Rabia Hameed, and Jihad Younis.

DISCLOSURE STATEMENT

The authors declare that there are no conflicts of interest regarding the publication of this paper.

DATA AVAILABILITY STATEMENT

The data used to support the findings of the study are available within this paper.

REFERENCES

[1] A. Shahzad, F. Khan, A. Ghaffar, G. Mustafa, K.-S. Nisar, and D. Baleanu, "A novel numerical algorithm to estimate the subdivision depth of binary subdivision schemes," *Symmetry*, vol. 12, no. 1, pp. 1–66, Jan. 2020.

[2] A. Shahzad, F. Khan, A. Ghaffar, S.-W. Yao, M. Inc, and S. Ali, "A novel numerical method for computing subdivision depth of quaternary schemes," *Mathematics*, vol. 9, no. 8, pp. 789–809, Apr. 2021.

[3] A. Ghaffar, Z. Ullah, M. Bari, K.-S. Nisar, M.-M. Al-Qurashi, and D. Baleanu, "A new class of $2m$-point binary non-stationary subdivision schemes," *Adv. Difference Equ.*, vol. 2019, Dec. 2019, Art. no. 325.

[4] A. Nawaz, A. Ghaffar, F. Khan, and S.-A. A. Karim, "A new 7-point quaternary approximating subdivision scheme," *Intell. Syst. Model. Simul. II*, vol. 444, pp. 545–566, Oct. 2022.

[5] C. Deng and W. Ma, "Efficient evaluation of subdivision schemes with polynomial reproduction property," *J. Comput. Appl. Math.*, vol. 294, pp. 403–412, Mar. 2016.

[6] C. Deng, H. Xu, W. Ma, and Y. Li, "Repeated local operations and associated interpolation properties of dual $2n$-point subdivision schemes," *J. Comput. Appl. Math.*, vol. 349, pp. 344–353, Mar. 2019.

[7] C. Conti and S. Huning, "An algebraic approach to polynomial reproduction of Hermite subdivision schemes," *J. Comput. Appl. Math.*, vol. 349, pp. 302–315, Mar. 2019.

[8] G.-M. Chaikin, "An algorithm for high-speed curve generation," *Comput. Graph. Image Process.*, vol. 3, no. 4, pp. 346–349, Dec. 1974.

[9] G. de Rham, "Sur une courbe plane," *J. Math. Pares App.*, vol. 35, pp. 25–42, Oct. 1956.

[10] G. Mustafa and F. Khan, "A new 4-point quaternary approximating subdivision scheme," *Abstract Appl. Anal.*, vol. 2009, pp. 1–14, Jun. 2009.

[11] G. Mustafa, A. Ghaffar, and M. Bari, "$(2n-1)$-point binary approximating scheme," in *Proc. 8th Int. Conf. Digit. Inf. Manage. (ICDIM)*, Sep. 2013, pp. 363–368.

[12] G. Mustafa, M. Asghar, S. Ali, A. Qamar, and J.-B. Liu, "The family of multiparameter quaternary subdivision schemes," *J. Math.*, vol. 2021, pp. 1–12, Dec. 2021.

[13] G. Mustafa and R. Hameed, "Families of univariate and bivariate subdivision schemes originated from quartic B-spline," *Adv. Comput. Math.*, vol. 43, no. 5, pp. 1131–1161, Oct. 2017.

[14] J. Shi, J. Tan, Z. Liu, and L. Zhang, "Six-point subdivision schemes with cubic precision," *Math. Problems Eng.*, vol. 2018, pp. 1–9, Jan. 2018.

[15] J. Tan, Y. Yao, H. Cao, and L. Zhang, "Convexity preservation of 5-point binary subdivision scheme with a parameter," *Appl. Math. Comput.*, vol. 245, pp. 279–288, Oct. 2014.

[16] J. Pana, S. Lin, and X. Luoa, "A combined approximating and interpolating subdivision scheme with C^2 continuity," *Appl. Math. Lett.*, vol. 25, no. 12, pp. 2140–2148, Dec. 2012.

[17] M. Bari, R. Bashir, and G. Mustafa, "$3n$-point quaternary shape preserving subdivision schemes," *Mehran Univ. Res. J. Eng. Technol.*, vol. 36, no. 3, pp. 489–500, Jul. 2017.

[18] M. Bari, G. Mustafa, and A. Hussain, "Fractal generation by 7-point binary approximating subdivision scheme," *Int. J. Appl. Phys. Math.*, vol. 8, no. 4, pp. 105–111, Oct. 2018.

[19] N. Dyn and D. Levin, "Stationary and non-stationary binary subdivision schemes," in *Computer Aided Geometric Design*, T. Lyche and L. Schumaker, Eds. New York, NY, USA: Academic, Jan. 1992, pp. 209–216.

[20] N. Dyn, J.-A. Gregory, and D. Levin, "Analysis of uniform binary subdivision schemes for curve design," *Construct. Approx.*, vol. 7, pp. 127–147, Dec. 1991.

[21] N. Dyn and D. Levin, "Subdivision schemes in geometric modelling," *Acta Numerica*, vol. 11, pp. 73–144, Jan. 2002.

[22] P. Ashraf, G. Mustafa, A. Ghaffar, R. Zahra, K.-S. Nisar, E.-E. Mahmoud, and W.-R. Alharbi, "Unified framework of approximating and interpolatory subdivision schemes for construction of class of binary subdivision schemes," *J. Function Spaces*, vol. 2020, pp. 1–12, Dec. 2020.

[23] R. Donat and D.-F. Yanez, "A nonlinear Chaikin-based binary subdivision scheme," *J. Comput. Appl. Math.*, vol. 349, pp. 379–389, Mar. 2019.

[24] R. Hameed and G. Mustafa, "Construction and analysis of binary subdivision schemes for curves and surfaces originated from Chaikin points," *Int. J. Anal.*, vol. 2016, pp. 1–15, Oct. Oct. 2016.

[25] R. Hameed and G. Mustafa, "Family of *a*-point *b*-ary subdivision schemes with bell-shaped mask," *Appl. Math. Comput.*, vol. 309, pp. 289–302, Sep. 2017.

[26] R.-J. Zhang, "A few conjectures on a four-point interpolatory subdivision scheme," *Comput. Aided Geometric Des.*, vol. 82, pp. 101925–101933, Oct. 2020.

[27] S. Hashmi and G. Mustafa, "Estimating error bounds for quaternary subdivision schemes," *J. Math. Anal. Appl.*, vol. 358, no. 1, pp. 159–167, Oct. 2009.

[28] S.-S. Siddiqi and N. Ahmad, "A new three-point approximating C^2 subdivision scheme," *Appl. Math. Lett.*, vol. 20, no. 6, pp. 707–711, Jun. 2007.

[29] S.-S. Siddiqi and K. Rehan, "Improved binary 4-point subdivision scheme and new corner cutting scheme," *Comput. Math. With Appl.*, vol. 59, no. 8, pp. 2647–2657, Apr. 2010.

[30] S.-S. Siddiqi and M. Younis, "Construction of *m*-point binary approximating subdivision schemes," *Appl. Math. Lett.*, vol. 26, no. 3, pp. 337–343, Mar. 2013.

[31] S.-S. Siddiqi and M. Younis, "The *m*-point quaternary approximating subdivision schemes," *Amer. J. Comput. Math.*, vol. 3, pp. 1–5, Mar. 2013.

[32] Y.-X. Hao, R.-H. Wang, and C.-J. Li, "Analysis of a 6-point binary subdivision scheme," *Appl. Math. Comput.*, vol. 218, no. 7, pp. 3209–3216, Dec. 2011.

SIDRA NOSHEEN received the master's degree in mathematics, the B.Ed. degree, the M.Ed. degree, and the M.Phil. degree in computational mathematics from The Islamia University of Bahawalpur, Pakistan, in 2010, 2011, 2012, and 2016, respectively. Currently, she is pursuing the Ph.D. degree with the Department of Mathematics, The Government Sadiq College Women University, Bahawalpur.

RABIA HAMEED received the M.Sc. degree in pure mathematics and the M.Phil. and Ph.D. degrees in computational mathematics from The Islamia University of Bahawalpur, Pakistan, in 2008, 2011, and 2018, respectively. Since March 2017, she has been held the position of a Lecturer with the Department of Mathematics, The Government Sadiq College Women University, Bahawalpur. Her scholarly contributions include numerous publications in prestigious journals recognized by the Higher Education Commission (HEC). Her research interests include numerical computing, approximation theory, and computer-aided geometric design (CAGD).

JIHAD YOUNIS received the bachelor's, master's, and Ph.D. degrees from the Department of Mathematics, Aden University, Yemen. He is currently a Lecturer with the Department of Mathematic, Aden University. With a prolific academic career, he has collaborated on more than 70 scientific research articles published in esteemed international journals. His primary research interests include special functions and fractional calculus.