Punjab University
Journal of Mathematics (ISSN 1016-2526)
Vol. 50(3)(2018) pp. 25-42

Stationary and Non-Stationary Univariate Subdivision Schemes

Muhammad Asghar
Department of Mathematics,
The Islamia University of Bahawalpur, Pakistan.
Email: m.asgharmaths@gmail.com

Ghulam Mustafa
Department of Mathematics,
The Islamia University of Bahawalpur, Pakistan.
Email: Corresponding authors: ghulam.mustafa@iub.edu.pk

Received: 16 June, 2017 / Accepted: 08 January, 2018 / Published online: 10 April, 2018

Abstract. The generalized symbols for family of b-ary $(b \geq 2)$, univariate stationary and non-stationary subdivision schemes have been presented. These symbols are based on Lane-Riesenfeld algorithm. In binary case, uniform B-splines schemes, Hormann and Sabin family, and Novara and Romani family of schemes can be derived from our schemes. In higher arity case, we present the analysis of proposed family in stationary context.

AMS (MOS) Subject Classification Codes: 65D17, 65D07, 65D05.
Key Words: Higher arity; stationary and non-stationary subdivision scheme; polynomial generation and reproduction; conic reproduction.

1. Introduction

Computer Aided Geometric Design (CAGD) deals with the mathematical description of shapes for use in computer graphics, numerical analysis and approximation theory. The important tool of CAGD is subdivision schemes. Subdivision schemes are iterative formulas for generation of smooth curves and surfaces. In recent years, subdivision schemes have become an integral part of computer graphics in view of their extensive variety of applications in the field of visualizations, animation and image processing. If the mask of subdivision schemes are dependent on subdivision level k then subdivision schemes are called non-stationary otherwise it is said to be stationary.

Initially Lane and Riesenfeld presented an algorithm [14] for subdividing uniform B-splines schemes of order l, with $l \in \mathbb{N}$. After that, this algorithm is used in different variants [5]. The symbol of Dubuc-Deslauriers [9] schemes are also containing the symbol of uniform B-splines subdivision schemes. The family of subdivision schemes presented by Hormann and Sabin [12] is also based on Lane and Riesenfeld algorithm. Ashraf et al, [1] also used this algorithm for generation of new family of binary subdivision schemes. Mustafa and Hameed [16] presented parametric family of binary approximating subdivision schemes using Lane-Riesenfeld algorithm. Mustafa and Bari [17] presented a family of subdivision schemes for fitting data. Mustafa et al. [15] presented bivariate subdivision schemes based on cubic polynomial.

In non-stationary context, Conti and Romani [5, 8] presented the mask of stationary and non-stationary B-spline subdivision schemes. They also presented the conditions for exponential polynomial reproduction using non-stationary subdivision schemes. Novara and Romani [18] offered two new families of subdivision schemes and its non-stationary version using Lane-Riesenfeld algorithm.

1.1. **Motivation.** After this literature, questions arise in our mind like: Can we generally use Lane-Riesenfeld algorithm for higher arity? Are we able to make a general formula which extend Lane-Riesenfeld algorithm in both stationary and non-stationary univariate schemes? Higher arity subdivision schemes gives better smoothness as compare to the lower arity subdivision schemes. These prompted us to answer these questions.

In this paper, we offer Lane-Riesenfeld algorithm for higher arity schemes. We also present general symbols for both stationary and non-stationary, univariate subdivision schemes. Our symbols are able to reproduce many family of subdivision schemes. Many existing binary univariate subdivision schemes are presented in a unified way. Uniform B-splines schemes, Hormann and Sabin [12] and Novara and Romani [18] family of schemes are generated from these algorithms. In higher arity case, we give the complete analysis of family of uniform B-splines schemes, family of stationary and non-stationary schemes.

The paper is unfolded as follows. Section 2 is for all the basic concepts of stationary and non-stationary subdivision schemes. Section 3 is devoted for stationary Lane-Riesenfeld algorithm. In Section 4, the non-stationary version of Lane-Riesenfeld algorithm is presented. Section 5 is devoted for tensor product formulas. Visual performance and conclusions are drawn in Sections 6 and 7 respectively.

2. PRELIMINARIES

This section contains all basic concepts of stationary and non-stationary subdivision schemes that we will use in rest of the paper.

Definition 2.1. [18] ***Basic concepts of stationary scheme:*** *If the refinement rules remain same in all levels of refinement then it is called stationary scheme. The refinement rules of stationary subdivision schemes are*

$$f^{k+1}_{bi+h} = \sum_{j\,\in\,\mathbb{Z}} a_{bj+h} f^{k}_{i-j}, h = 0, 1, 2 \cdots (b-1), \tag{2. 1}$$

where b is any integer $(b \geq 2)$*, the set of coefficients* $a_{bj+h} \in \mathbb{R}, i \in \mathbb{Z}$*,* $k \geq 0$ *appearing in (2.1) is called mask of subdivision schemes* S_a *and is denoted by* a*. The* z*-transform of the mask* $a = \{a_i : i \,\in\, \mathbb{Z}\}$ *of the scheme can be defined as*

$$a(z) = \sum_{i\,\in\,\mathbb{Z}} a_i z^i, \tag{2. 2}$$

which is also called Laurent polynomial or symbol of the scheme. This symbol is used to investigate the properties of subdivision scheme, such as convergence, smoothness, polynomial reproduction degree, polynomial generation degree, parametrization and Hölder regularity.

Definition 2.2. [18] ***Basic concepts of non-stationary scheme:*** *If the refinement rules are not same in all levels of refinement then it is called non-stationary scheme. The refinement rules of non-stationary subdivision schemes are*

$$f_{bi+h}^{k+1} = \sum_{j\,\in\,\mathbb{Z}} a_{bj+h}^{k} f_{i-j}^{k}, h = 0, 1, 2 \cdots (b-1), \tag{2. 3}$$

where b is any integer $(b \geq 2)$, the set of coefficients $a_{bj+h}^{k} \in \mathbb{R}, i \in \mathbb{Z}$ appearing in (2.3) is called mask of non-stationary subdivision schemes S_{a^k} and is denoted by a^k. The z-transform of the mask $a^k = \{a_i^k : i \in \mathbb{Z}\}$, of the scheme can be given as

$$a^k(z) = \sum_{i\,\in\,\mathbb{Z}} a_i^k z^i. \tag{2. 4}$$

Stationary subdivision schemes S_a and a non-stationary subdivision schemes $S_{a^k}, k \in N_0$, are said to be asymptotically equivalent [4, 7] *if their masks satisfy*

$$\lim_{k\to+\infty} a^k = a. \tag{2. 5}$$

We can easily analyze the properties of non-stationary scheme using (2. 5) and (2. 2).

3. Stationary Lane-Riesenfeld algorithm

Lane-Riesenfeld algorithm is a two step algorithm one is refining and second is smoothing step. Refining step is used to insert new points on the initial control polygon and smoothing step is used to modifies the obtained points. The limit curves are obtained by using successive iterations of smoothing steps.

3.1. **Construction and analysis of generalized B-splines symbols.** A generalized B-splines symbol for any arity $(b \geq 2)$ has been presented using the well known Lane-Riesenfeld algorithm. This algorithm is based on smoothing operator described by a symbol of the form

$$S_b(z) = \frac{(1 + z + z^2 + \cdots + z^{b-1})}{b}, \tag{3. 6}$$

and refining factor is defined as

$$R_b(z) = \frac{(1 + z + z^2 + \cdots + z^{b-1})^2}{b} z^{-(b-1)}. \tag{3. 7}$$

By applying Lane-Riesenfeld algorithm on smoothing operator $S_b(z)$ and refining operator $R_b(z)$, we have

$$A_n^b(z) = z^{-[\frac{n}{b}]}(S_b(z))^n R_b(z). \tag{3. 8}$$

After substituting the values of $S_b(z)$ and $R_b(z)$ we get,

$$A_n^b\,(z) = \frac{(1 + z + z^2 + \cdots + z^{b-1})^{n+2}}{b^{n+1} z^{(b-1)+[\frac{n}{b}]}}, \tag{3. 9}$$

which is the general symbol of $(n+1)$th degree polynomial B-spline, where b is any arity $(b \geq 2)$.
By substituting $b = 2$ and $n = 1$, we get the Laurent polynomial of well known 2-point chainkin's [2] corner cutting scheme

$$A_1^2(z) = \frac{1}{4}(z^{-1} + 3 + 3z + z^2).$$

Mask of the scheme is $\frac{1}{4}[1, 3, 3, 1]$. The scheme corresponding to the mask is

$$\begin{aligned} f_{2i}^{k+1} &= \frac{1}{4}\left[3f_i^k + f_{i+1}^k\right], \\ f_{2i+1}^{k+1} &= \frac{1}{4}\left[f_i^k + 3f_{i+1}^k\right]. \end{aligned}$$

By substituting $b = 3$ and $n = 2$, we get the Laurent polynomial of 3-point ternary scheme

$$A_2^3(z) = \frac{1}{27}(z^{-2} + 4z^{-1} + 10 + 16z + 19z^2 + 16z^3 + 10z^4 + 4z^5 + z^6).$$

Mask of the scheme is $\frac{1}{27}[1, 4, 10, 16, 19, 16, 10, 4, 1]$. The scheme corresponding to the mask is

$$\begin{aligned} f_{3i}^{k+1} &= \frac{1}{27}\left[10f_{i-1}^k + 16f_i^k + f_{i+1}^k\right], \\ f_{3i+1}^{k+1} &= \frac{1}{27}\left[4f_{i-1}^k + 19f_i^k + 4f_{i+1}^k\right], \\ f_{3i+2}^{k+1} &= \frac{1}{27}\left[f_{i-1}^k + 16f_i^k + 10f_{i+1}^k\right]. \end{aligned}$$

Similarly when $b = 4$ and $n = 3$, we get the Laurent polynomial of 4-point quaternary scheme

$$\begin{aligned} A_3^4(z) = &\frac{1}{256}(z^{-3} + 5z^{-2} + 15z^{-1} + 35 + 65z + 101z^2 + 135z^3 + 155z^4 + 155z^5 \\ &+135z^6 + 101z^7 + 101z^8 + 65z^9 + 35z^{10} + 15z^{11} + z^{12}). \end{aligned}$$

Mask of the scheme is $\frac{1}{256}[1, 5, 15, 35, 65, 101, 135, 155, 155, 135, 101, 65, 35, 15, 5, 1]$. The scheme corresponding to the mask is

$$\begin{aligned} f_{4i}^{k+1} &= \frac{1}{256}\left[35f_{i-1}^k + 155f_i^k + 65f_{i+1}^k + f_{i+2}^k\right], \\ f_{4i+1}^{k+1} &= \frac{1}{256}\left[15f_{i-1}^k + 135f_i^k + 101f_{i+1}^k + 5f_{i+2}^k\right], \\ f_{4i+2}^{k+1} &= \frac{1}{256}\left[5f_{i-1}^k + 101f_i^k + 135f_{i+1}^k + 15f_{i+2}^k\right], \\ f_{4i+3}^{k+1} &= \frac{1}{256}\left[f_{i-1}^k + 65f_i^k + 155f_{i+1}^k + 35f_{i+2}^k\right]. \end{aligned}$$

In Table 1, we present arity, complexity, support and mask of the proposed schemes corresponding to $n = 0, 1, 2, 3, 4$ and 5.

TABLE 1. *Shows the arity, complexity, support and mask of the schemes corresponding to the different values of n, here m shows complexity of the schemes (i.e. 2-, 3-, ...-point schemes), while b, S and A_n^b stand for arity, support width and mask of the schemes respectively.*

n	b	m	S	Mask
0	2	2	2	$A_0^2 = \frac{1}{2}[1, 2, 1]$
1	2	2	3	$A_1^2 = \frac{1}{4}[1, 3, 3, 1]$
2	2	3	4	$A_2^2 = \frac{1}{8}[1, 4, 6, 4, 1]$
3	2	3	5	$A_3^2 = \frac{1}{16}[1, 5, 10, 10, 5, 1]$
4	2	4	6	$A_4^2 = \frac{1}{32}[1, 6, 15, 20, 15, 6, 1]$
5	2	4	7	$A_5^2 = \frac{1}{64}[1, 7, 21, 35, 35, 21, 7, 1]$
0	3	2	4	$A_0^3 = \frac{1}{3}[1, 2, 3, 2, 1]$
1	3	2	6	$A_1^3 = \frac{1}{9}[1, 3, 6, 7, 6, 3, 1]$
2	3	3	8	$A_2^3 = \frac{1}{27}[1, 4, 10, 16, 19, 16, 10, 4, 1]$
3	3	4	10	$A_3^3 = \frac{1}{81}[1, 5, 15, 30, 45, 51, 45, 30, 15, 5, 1]$
4	3	5	12	$A_4^3 = \frac{1}{243}[1, 6, 21, 50, 90, 126, 141, 126, 90, 50, 21, 6, 1]$
5	3	5	14	$A_5^3 = \frac{1}{729}[1, 7, 28, 77, 161, 266, 357, 393, 357, 266, 161, 77, 28, 7, 1]$
0	4	2	6	$A_0^4 = \frac{1}{4}[1, 2, 3, 4, 3, 2, 1]$
1	4	3	9	$A_1^4 = \frac{1}{16}[1, 3, 6, 10, 12, 12, 10, 6, 3, 1]$
2	4	4	12	$A_2^4 = \frac{1}{64}[1, 4, 10, 20, 31, 40, 44, 40, \cdots 4, 1]$
3	4	4	15	$A_3^4 = \frac{1}{256}[1, 5, 15, 35, 65, 101, 135, 155, 155, 135, \cdots, 5, 1]$
4	4	5	18	$A_4^4 = \frac{1}{1024}[1, 6, 21, 56, 120, 216, 336, 456, 546, 580, 546, \cdots, 6, 1]$
5	4	6	21	$A_5^4 = \frac{1}{4096}[1, 7, 28, 84, 203, 413, 728, 1128, 1554, 1918, 2128, 2128, \cdots, 7, 1]$
0	5	2	8	$A_0^5 = \frac{1}{5}[1, 2, 3, 4, 5, 4, 3, 2, 1]$
1	5	3	12	$A_1^5 = \frac{1}{25}[1, 3, 6, 10, 15, 18, 19, 18, 15, 10, 6, 3, 1]$
2	5	4	16	$A_2^5 = \frac{1}{125}[1, 4, 10, 20, 35, 52, 68, 80, 85, 80, 68, 52, 35, 20, 10, 4, 1]$
3	5	5	20	$A_3^5 = \frac{1}{625}[1, 5, 15, 35, 70, 121, 185, 255, 320, 365, 320, \cdots, 5, 1]$
4	5	5	24	$A_4^5 = \frac{1}{3125}[1, 6, 21, 56, 126, 246, 426, 666, 951, 1246, 1506, 1686, 1751, 1686, \cdots, 6, 1]$
5	5	6	28	$A_5^5 = \frac{1}{15625}[1, 7, 28, 84, 210, 455, 875, 1520, 2415, 3535, 4795, 6055, 7140, 7875,$
				$8135, 7875, \cdots, 7, 1]$
0	6	2	10	$A_0^6 = \frac{1}{6}[1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1]$
1	6	3	15	$A_1^6 = \frac{1}{36}[1, 3, 6, 10, 15, 21, 25, 27, 27, \cdots, 3, 1]$
2	6	4	20	$A_2^6 = \frac{1}{216}[1, 4, 10, 20, 35, 56, 80, 104, 125, 140, 146, 140, \cdots, 4, 1]$
3	6	5	25	$A_3^6 = \frac{1}{1296}[1, 5, 15, 35, 70, 126, 205, 305, 420, 540, 651, 735, 780, 780, \cdots, 5, 1]$
4	6	6	30	$A_4^6 = \frac{1}{7776}[1, 6, 21, 56, 126, 252, 456, 756, 1161, 1666, 2247, 2856, 3431,$
				$3906, 4221, 4332, 4221, \cdots, 6, 1]$
5	6	6	35	$A_5^6 = \frac{1}{46656}[1, 7, 28, 84, 210, 462, 917, 1667, 2807, 4417, 6538, 9142, 12117, 15267,$
				$18327, 20993, 22967, 24017, 24017, \ldots, 7, 1]$

Analysis of A_n^b schemes: Aim of this section is to present the analysis of proposed A_n^b family of schemes. Laurent polynomial (symbol) method [10] is used to compute the continuity, degree of generation, degree of reproduction of the A_n^b-schemes. While Riouls method [19] is used to compute lower and upper bounds on Hölder continuity and exact

Hölder continuity can also be computed by using Floater and Muntingh algorithm [11]. Support of the schemes are computed by [13]. For family of binary schemes, we put $b = 2$ in (3.4). The analysis of family of binary schemes are presented in [12]. Here we present the general analysis of proposed family.

Theorem 3.2. *The schemes corresponding to symbol A_n^b is C^n continuous.*

Proof. From (3. 10), we have

$$A_n^b\,(z) = \left(\frac{1+z+z^2+\cdots+z^{b-1}}{b}\right)^{n+1} b(z), \tag{3. 10}$$

where $b(z) = \frac{1+z+z^2+\cdots+z^{b-1}}{z^{(b-1)+[\frac{n}{b}]}}$. Let S_a and S_b be the schemes corresponding to the symbols $A_n^b\,(z)$ and $b(z)$ respectively. Since

$$\left\|\frac{1}{b}S_b\right\|_\infty = \left(\frac{1}{b}\right)\max\left\{\sum_{j\in\mathbb{Z}}|b_{bj}|,\sum_{j\in\mathbb{Z}}|b_{bj+1}|,\sum_{j\in\mathbb{Z}}|b_{bj+2}|,\cdots\sum_{j\in\mathbb{Z}}|b_{bj+(b-1)}|\right\}<1.$$

Therefore by [10], the scheme S_a is C^n continuous. □

Theorem 3.3. *Hölder regularity of schemes corresponding to the symbol A_n^b is $n+1$.*

Proof. From (3. 10) we have $b_0 = b_1 =, \cdots, = b_{b-1} = 1, k = n+1, m = b-1$ and thus $q = 0, 1, 2, \cdots b-1$ and $B_0, B_1, \cdots B_{b-1}$ are the matrices with elements $(B_q)_{i,j} = b_{b-1+i-b\,j+q}$. By [11] the Hölder regularity is given by $r = k - \log_b(\mu)$, where μ is the joint spectral radius of the matrices $B_0, B_1, \cdots, B_{b-1}$, that is, $\mu = \rho(B_0, B_1, \cdots, B_{b-1})$. For bounds on Hölder regularity we calculate $max\,\{\rho(B_0), \rho(B_1), \cdots, \rho(B_{b-1}),\} \leq \mu \leq max\,\{\|(B_0)\|_\infty, \|(B_1)\|_\infty, \cdots, \|(B_{b-1})\|_\infty\}$. Since μ is bounded from below by the spectral radii and from above by the norm of the metrics $B_0, B_1, \cdots, B_{b-1}$. So $max\{1, 1, \cdots, 1\} = \mu = max\{1, 1, \cdots, 1\}$. This implies $\mu = 1$, the exact Hölder regularity of the scheme corresponding to the symbol A_n^b is $r = n+1-\log_b(1) = n+1$. □

Theorem 3.4. *Generation degree of the scheme corresponding to the symbol A_n^b is $n+1$.*

Proof. The Laurent polynomial of the scheme defined in (3. 10) can be written as

$$A_n^b\,(z) = \left(\frac{1+z+z^2+\cdots+z^{b-1}}{b}\right)^{n+2} b(z),$$

where $b(z) = \frac{b(1+z+z^2+\cdots+z^{b-1})}{z^{(b-1)+[\frac{n}{b}]}}$. Hence by [10], generation degree is $n+1$. □

Theorem 3.5. *The schemes corresponding to the symbol A_n^b have linear reproduction and parameterizations depends on n and b.*

Proof. By taking the first derivative of (3. 10) and putting $z = 1$, we get $\left(A_n^b\right)'(1) = (n+2)\sum_{i=1}^{b-1}(i)$. This implies that $\tau = \frac{(n+2)\sum_{i=1}^{b-1}(i)}{b}$, so the parametrization of scheme corresponding to the symbol $A_n^b\,(z)$ depends on b and n. All even arity schemes have dual parametrization for odd values of n and primal parametrization for even values of n. All odd arity subdivision schemes have primal parametrization. We can easily verify that

$\left(A_n^b\right)^k\left(\alpha_b^j\right)=0$, where $\alpha_b^j=e^{\left(\frac{2\pi i}{b}\right)j}, j=1,2,\cdots(b-1)$ and $\left(A_n^b\right)^k(1)=b\prod_{l=0}^{k-1}(\tau-l)$ for $k=0$ and 1. Then by [10], the scheme corresponding to the symbol A_n^b have linear reproduction and parameterizations depends on n and b. This completes the proof. $\square$

3.6. **Construction of family of b-ary $(b\geq 2)$ schemes.** Here we present a general symbol which generates different families of subdivision schemes like Hormann-Sabin family and many other families of b-ary subdivision schemes. This algorithm is defined as a combination of generalized B-spline with the kernel

$$H_n^b\left(z\right)=A_n^b(z)K_n\left(z\right), \tag{3. 11}$$

where $A_n^b(z)$ is defined in (3.4) and $K_n(z)$ from [12] is defined as

$$K_n\left(z\right)=-\frac{n+2}{8}z^{-1}+\frac{n+6}{4}-\frac{n+2}{8}z. \tag{3. 12}$$

Hormann and Sabin's [12] proposed a family of stationary subdivision schemes with cubic precision to increase the degree of polynomial reproduction of B-splines scheme. This family of schemes is special case of our proposed algorithm. If we put $b=2$ in (3.11), we get the polynomial

$$H_n^2\left(z\right)=\frac{(1+z)^{n+2}}{2^{n+1}z^{1+[\frac{n}{2}]}}\left(-\frac{n+2}{8}z^{-1}+\frac{n+6}{4}-\frac{n+2}{8}z\right). \tag{3. 13}$$

By substituting the different values of n, we get the family members of Hormann and Sabin family. In Table 2, we present the results of H_n^b family. In general proposed family $H_n^b\left(z\right)$ defined in (3. 11) is C^n continuous, generation degree is $n+1$, the family of schemes have linear reproduction w.r.t primal parametrization for even n and dual parametrization for odd n.

TABLE 2. *Here we present analysis of proposed families of H_n^b schemes for $b=2,3,4$ and 5. Here b, C, G_d, R_d, τ, and P are arity, continuity, generation degree, reproduction degree, shift parameter and parametrization.*

b	C	G_d	R_d	τ	P
2	C^n	n+1	3	$\tau=\frac{n+2}{2}$	Primal for even n
					Dual for odd n
3	C^n	n+1	1	$\tau=n+2$	Dual $\forall\quad n$
4	C^n	n+1	1	$\tau=\frac{6n+12}{4}$	Primal for even n
					Dual for odd n
5	C^n	n+1	1	$\tau=2n+4$	Dual $\forall\quad n$

4. NON-STATIONARY LANE-RIESENFELD ALGORITHM

In this section, we present non-stationary version of Lane-Riesenfeld algorithm. Let

$$\upsilon^k=\frac{1}{2}\left(e^{i\frac{t}{2^{k+1}}}+e^{-i\frac{t}{2^{k+1}}}\right)=\cos\left(\frac{t}{2^{k+1}}\right),\quad \text{with}\quad t\in[0,\pi)\cup i\mathbb{R}^+.$$

Define an arbitrary $\upsilon^0 \in (0, +\infty)$ as

$$\upsilon^0 = \cos\left(\frac{t}{2}\right) = \begin{cases} \cos\left(\frac{\alpha}{2}\right) \in (0,1) & \text{if} \quad t = \alpha, \quad \alpha \in (0,\pi), \\ 1 & \text{if} \quad t = 0, \\ \cosh\left(\frac{\alpha}{2}\right) \in (1, +\infty) & \text{if} \quad t = i\alpha, \quad \alpha \in \mathbb{R}^+. \end{cases}$$

We also define

$$\upsilon^{k+1} = \left(\frac{1+\upsilon^k}{2}\right)^{\frac{1}{2}},$$

remember that if $t = 0$ then $\upsilon^k = \upsilon^{k+1} = 1$, similarly we have

$$\lim_{k\to+\infty} \upsilon^k = \lim_{k\to+\infty} \upsilon^{k+1} = 1. \tag{4. 14}$$

4.1. **Non-stationary generalized B-splines symbol.** In non-stationary context, the generalized B-splines algorithm is described using smoothing factor defined by

$$S_b^k(z) = \frac{(1+z+z^2+\cdots+z^{b-1})}{b\upsilon^{k+1}}, \tag{4. 15}$$

and refining factor defined by

$$R_b^k(z) = \frac{(1+z+z^2+\cdots+z^{b-1})^2}{b\upsilon^k} z^{-(b-1)} + \left(1 - \frac{1}{\upsilon^k}\right).$$

The non-stationary generalized B-splines algorithm is defined by n successive applications of the smoothing operator followed by one application of the refine operator at k-level symbol

$$A_{b,n}^k(z) = z^{-[\frac{n}{b}]}(S_b^k(z))^n R_b^k(z). \tag{4. 16}$$

In case of binary scheme, we put $b = 2$ in (4. 16) and after simplification, we obtain the general formula of non-stationary binary B-splines schemes [18].

$$A_{2,n}^k(z) = \frac{(1+z)^n(z+2\upsilon^k+z^{-1})}{2\upsilon^k z^{[\frac{n}{2}]}(2(1+\upsilon^k))^{\frac{n}{2}}}. \tag{4. 17}$$

Proposition 4.2. *Non-stationary Lane-Riesenfeld algorithm is equivalent to the stationary Lane-Riesenfeld algorithm*

$$\lim_{k\to+\infty} A_{b,n}^k(z) = A_n^b(z).$$

Proof. We can easily verify the above result using

$$\lim_{k\to+\infty} S_b^k(z) = S_b(z) \quad \text{and} \quad \lim_{k\to+\infty} R_b^k(z) = R_b(z).$$

□

4.3. **Non-stationary family of b-ary $(b \geq 2)$ schemes.** In this section, we present an algorithm for non-stationary family of b-ary schemes for $b \geq 2.$ This algorithm is defined as

$$H_{b,n}^{k}(z) = A_{b,n}^{k}(z)K_{n}^{k}(z) + T_{b}^{k}(z), \tag{4. 18}$$

where $A_{b,n}^{k}(z)$ is defined in (4.16) and $T_{b}^{k}(z)$ is defined as

$$T_{b}^{k}(z) = \frac{(b-2)(\upsilon^{k}-1)(1+z+z^{2}+\cdots+z^{b-1})}{b(\upsilon^{k}+1)}. \tag{4. 19}$$

$T_{b}^{k}(z)$ is used for higher arity non-stationary subdivision schemes and $K_{n}^{k}(z)$ from [18] is defined as

$$K_{n}^{k}(z) = u_{n}z + (1-2u_{n}\upsilon^{k}) + u_{n}z^{-1}, \tag{4. 20}$$

with u_n is defined as

$$u_{n} = \frac{1}{2(\upsilon^{k}-1)} - \frac{\upsilon^{k}(\upsilon^{k+1})^{n}}{((\upsilon^{k})^{2}-1)}, \tag{4. 21}$$

which is the general algorithm of non-stationary family of any arity subdivision schemes. If we substitute $b = 2$ in (4. 18), we have Novara-Romani family [18] i.e.

$$\begin{aligned} H_{2,n}^{k}(z) \quad &= \quad z^{-[\frac{n}{2}]}\left(\frac{1+z}{2\upsilon^{k+1}}\right)^{n}\left(\frac{(1+z)^{2}}{2\upsilon^{k}}z^{-1} + \left(1-\frac{1}{\upsilon^{k}}\right)\right) \\ &\quad \times \left(u_{n}z + (1-2u_{n}\upsilon^{k}) + u_{n}z^{-1}\right). \end{aligned} \tag{4. 22}$$

By substituting $b = 3$ in (4.18), we get general symbol of family of ternary non-stationary subdivision schemes,

$$\begin{aligned} H_{3,n}^{k}(z) \quad &= \quad z^{-[\frac{n}{3}]}\left(\frac{1+z+z^{2}}{3\upsilon^{k+1}}\right)^{n}\left(\frac{(1+z+z^{2})^{2}}{3\upsilon^{k}}z^{-3} + \left(1-\frac{1}{\upsilon^{k}}\right)\right) \\ &\quad \times \left(u_{n}z + (1-2u_{n}\upsilon^{k}) + u_{n}z^{-1}\right) + \frac{\left(\upsilon^{k}-1\right)\left(1+z+z^{2}\right)}{3\left(\upsilon^{k}+1\right)}, \end{aligned}$$

After substituting the different values of n, we get family members of ternary non-stationary subdivision schemes.

Lemma 4.4. *For all $n \in \mathbb{N}$ and $\forall\, \upsilon^{0} \in (0, +\infty)$, the parameter u_{n}^{k} verifies*

$$\lim_{k\to+\infty} u_{n} = -\frac{n}{8} - \frac{1}{4}.$$

Proof. As we know that

$$u_{n} = \frac{1}{2(\upsilon^{k}-1)} - \frac{\upsilon^{k}(\upsilon^{k+1})^{n}}{((\upsilon^{k})^{2}-1)},$$

after simplification we get

$$u_{n} = \frac{(\upsilon^{k}+1) - \upsilon^{k}(\frac{\upsilon^{k}+1}{2})^{\frac{n}{2}}}{((\upsilon^{k})^{2}-1)}.$$

Now first apply the De l'Hopital rule and then use (4. 14) to get the required result. □

Proposition 4.5. *The symbol of non-stationary b-ary subdivision schemes with symbol at k-level is asymptotically equivalent to the symbol of stationary b-ary scheme*

$$\lim_{k\to+\infty} H^k_{b,n}(z) = H^b_n(z).$$

Proof. Using Lemma 4.2, we get

$$\lim_{k\to+\infty} K^k_n(z) = K_n(z).$$

From Proposition 4.1 and (4. 14), we obtain the required result. □

5. SURFACE CASE

In this section, we have presented the tensor product of pervious work. This section will consists of two subsections, one is for stationary tensor product and second is for non-stationary tensor product of the subdivision schemes.

5.1. **Tensor product of Lane-Riesenfeld algorithm.** A tensor product of Lane-Riesenfeld algorithm has been presented. Here we present two algorithms one is for the generalized B-spline stationary tensor product algorithm and second is for the family of higher arity stationary tensor product subdivision schemes. For this, smoothing factor is defined as

$$S_b(z_1,z_2) = \frac{(1+z_1+z_1^2+\cdots+z_1^{b-1})(1+z_2+z_2^2+\cdots+z_2^{b-1})}{b^2}, \tag{5. 23}$$

and the refining factor is defined as

$$R_b(z_1,z_2) = \frac{S(z_1z_2)(1+z_1+z_1^2+\cdots+z_1^{b-1})(1+z_2+z_2^2+\cdots+z_2^{b-1})}{z_1^{b-1}z_2^{b-1}}.$$

Simplified form of the refining factor is

$$R_b(z_1,z_2) = \frac{(1+z_1+z_1^2+\cdots+z_1^{b-1})^2(1+z_2+z_2^2+\cdots+z_2^{b-1})^2}{b^2z_1^{b-1}z_2^{b-1}}. \tag{5. 24}$$

So the tensor product of generalized B-spline algorithm is defined as

$$A^b_n(z_1,z_2) = z_1^{-[\frac{n}{b}]}z_2^{-[\frac{n}{b}]}(S_b(z_1z_2))^nR_b(z_1z_2).$$

Simplest form of the above algorithm is

$$A^b_n(z_1,z_2) = \frac{(1+z_1+z_1^2+\cdots+z_1^{b-1})^{n+2}(1+z_2+z_2^2+\cdots+z_2^{b-1})^{n+2}}{b^{2n+2}z_1^{b-1+[\frac{n}{b}]}z_2^{b-1+[\frac{n}{b}]}}, \tag{5. 25}$$

which is the general form of generalized B-splines tensor product algorithm for higher arity $(b \geq 2)$. If we substitute different values of b and n, we get the family members of higher arity B-splines tensor product subdivision schemes.
The tensor product algorithm for the family of higher arity subdivision schemes is defined as

$$H^b_n\left(z_1,z_2\right) = A^b_n\left(z_1,z_2\right)K_n\left(z_1,z_2\right), \tag{5. 26}$$

where $A^b_n\left(z_1,z_2\right)$ is defined in (5. 25) and $K_n\left(z_1,z_2\right)$ is defined as

$$K_n(z_1,z_2) = \left(-\frac{n+2}{8}z_1^{-1}+\frac{n+6}{4}-\frac{n+2}{8}z_1\right)\left(-\frac{n+2}{8}z_2^{-1}+\frac{n+6}{4}-\frac{n+2}{8}z_2\right).$$

This is the general form of tensor product family of higher arity subdivision schemes.

5.2. **Tensor product of non-stationary Lane-Riesenfeld algorithm.** The aim of this subsection is to present the tensor product of non-stationary Lane-Riesenfeld algorithm. This section contains two algorithms, one is for non-stationary tensor product generalized B-splines algorithm and other is for non-stationary tensor product of family of higher arity subdivision schemes. For tensor product of non-stationary Lane-Riesenfeld algorithm, non-stationary smoothing factor is defined as

$$S_b^k(z_1 z_2) = \frac{(1+z_1+z_1^2+\cdots+z_1^{b-1})(1+z_2+z_2^2+\cdots+z_2^{b-1})}{b^2(v^{k+1})^2}, \tag{5. 27}$$

and refining factor is defined as

$$R_b^k(z_1) = \frac{v^{k+1}}{v^k} S^k(z_1)(1+z_1+z_1^2+\cdots+z_1^{b-1})z_1^{-(b-1)} + \left(1-\frac{1}{v^k}\right).$$

The simplified form of $R_b^k(z_1)$ is

$$R_b^k(z_1) = \frac{(1+z_1+z_1^2+\cdots+z_1^{b-1})^2}{bv^k z_1^{(b-1)}} + \left(1-\frac{1}{v^k}\right).$$

Similarly $R_b^k(z_2)$ is

$$R_b^k(z_2) = \frac{(1+z_2+z_2^2+\cdots+z_2^{b-1})^2}{bv^k z_2^{(b-1)}} + \left(1-\frac{1}{v^k}\right).$$

The refining factor $R_b^k(z_1,z_2)$ for tensor product is

$$R_b^k(z_1,z_2) = R_b^k(z_1)R_b^k(z_2). \tag{5. 28}$$

The non-stationary tensor product generalized B-splines algorithm is defined as

$$A_n^k(z_1,z_2) = z_1^{-[\frac{n}{b}]} z_2^{-[\frac{n}{b}]} (S_b^k(z_1,z_2))^n R_b^k(z_1,z_2). \tag{5. 29}$$

This is the general form of non-stationary tensor product generalized B-splines algorithm. After substituting the different values of $b \geq 2$ and any positive integer n, we get the tensor product version of family of non-stationary B-splines schemes.

Lemma 5.3. *The smoothing-refining factor of non-stationary tensor product is asymptotically equivalent to the smoothing-refining factor of stationary tensor product*

$$\lim_{k\to+\infty} R_b^k(z_1,z_2) = R_b(z_1,z_2) \quad \textit{and} \quad \lim_{k\to+\infty} S_b^k(z_1,z_2) = S_b(z_1,z_2).$$

Proof. By using (2. 5) and Proposition 4.1, we get the results. □

Proposition 5.4. *The non-stationary tensor product of Lane-Riesenfelds algorithm is asymptotically equivalent to the stationary tensor product of Lane-Riesenfeld algorithm*

$$\lim_{k\to+\infty} A_n^k(z_1,z_2) = A_n^b(z_1,z_2).$$

Proof. The above result is straight forward by using Lemma 5.1. □

The algorithm of non-stationary tensor product of family of higher arity subdivision schemes is defined as

$$\begin{aligned} H_n^k(z_1, z_2) &= \left(z_1^{-[\frac{n}{b}]}\left(S_b^k(z_1)\right)^n R_b^k(z_1)K_n^k(z_1) + T_b^k(z_1)\right) \times \\ &\quad \left(z_2^{-[\frac{n}{b}]}\left(S_b^k(z_2)\right)^n R_b^k(z_2)K_n^k(z_2) + T_b^k(z_2)\right), \end{aligned}$$

where $S_b^k(z_1), S_b^k(z_2), R_b^k(z_1)$ and $R_b^k(z_2)$ are defined in previous section and

$$\begin{aligned} K_n^k(z_1) &= u_n z_1 + (1 - 2u_n v^k) + u_n z_1^{-1}, \\ K_n^k(z_2) &= u_n z_2 + (1 - 2u_n v^k) + u_n z_2^{-1}, \\ T_b^k(z_1) &= \frac{(b-2)(v^k - 1)(1 + z_1 + z_1^2 + \cdots + z_1^{b-1})}{b(v^k + 1)}, \\ T_b^k(z_2) &= \frac{(b-2)(v^k - 1)(1 + z_2 + z_2^2 + \cdots + z_2^{b-1})}{b(v^k + 1)}, \end{aligned}$$

with $u_n = \frac{1}{2(v^k-1)} - \frac{v^k(v^{k+1})^n}{((v^k)^2-1)}$, which is the general form of non-stationary tensor product family of higher arity algorithm.

Proposition 5.5. *The non-stationary tensor product family of higher arity algorithm is asymptotically equivalent to the stationary tensor product family of higher arity algorithm*

$$\lim_{k \to +\infty} H_n^k(z_1, z_2) = H_n^b(z_1, z_2).$$

Proof. As we know that

$$\lim_{k \to +\infty} K_n^k(z_1, z_2) = K_n(z_1, z_2), \quad \lim_{k \to +\infty} T_b^k(z_1) = 0 \quad \text{and} \quad \lim_{k \to +\infty} T_b^k(z_2) = 0.$$

By using Proposition 5.2, we get the required result. □

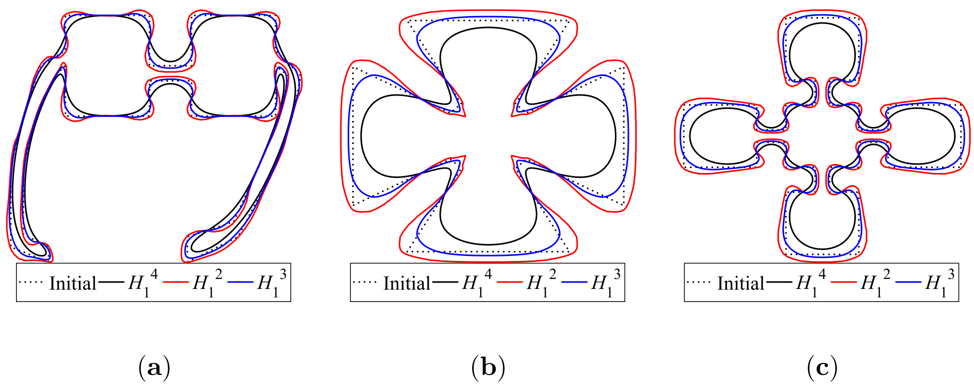
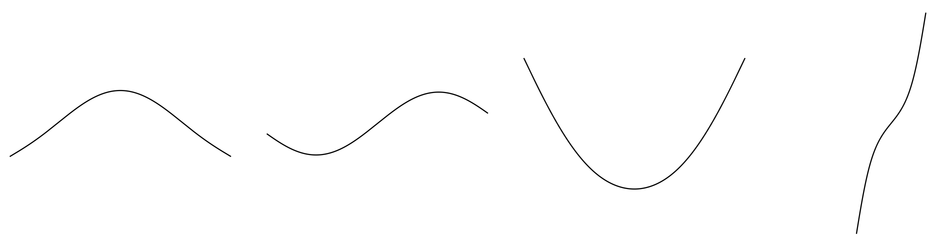


FIGURE 1. (**a**), (**b**) and (**c**) present limit curves for close polygons produced by schemes corresponding to H_1^2, H_1^3 and H_1^4.

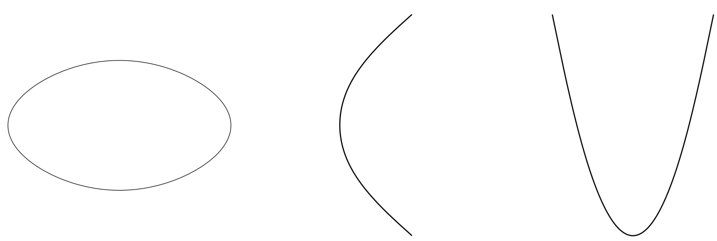
6. VISUAL PERFORMANCE

Here, we present the visual performance and comparison among proposed family members. Moreover the numerical reproduction of trigonometric and conic section by our proposed schemes are also presented. The control polygons are drawn by doted lines. The smooth curves are obtained by our proposed schemes. Figures 1 presents limit curves for close polygons generated by H_1^2, H_1^3 and H_1^4. Figures 1 shows that as arity increases, smoothness also increases. Figure 2 shows the numerical reproduction of trigonometric and hyperbolic functions by our proposed family of quaternary approximating schemes. Similarly, Figure 3 shows that the conic section can be reproduced numerically by our proposed family of quaternary approximating schemes. Figure 4 and Figure 5 shows the subdivision curves that contain the conic segments. Figure 6 and Figure 7 shows that the visual performance of tensor product surface subdivision schemes.



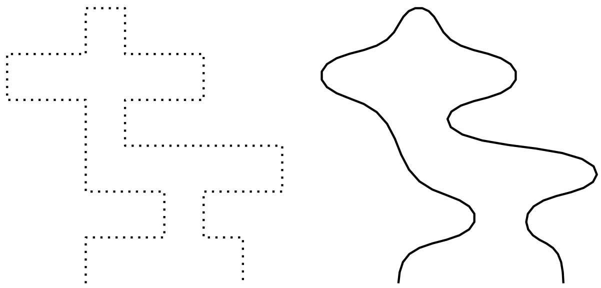
(**a**) $\cos(x)$ (**b**) $\sin(x)$ (**c**) $\cosh(x)$ (**d**) $\sinh(x)$

FIGURE 2. Reproduction of trigonometric and hyperbolic functions by scheme corresponding to A_3^4.



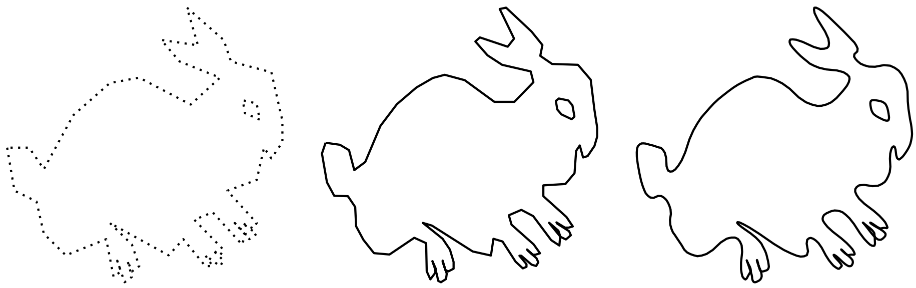
(**a**) (**b**) (**c**)

FIGURE 3. Reproduction of conics by scheme corresponding to A_3^4.

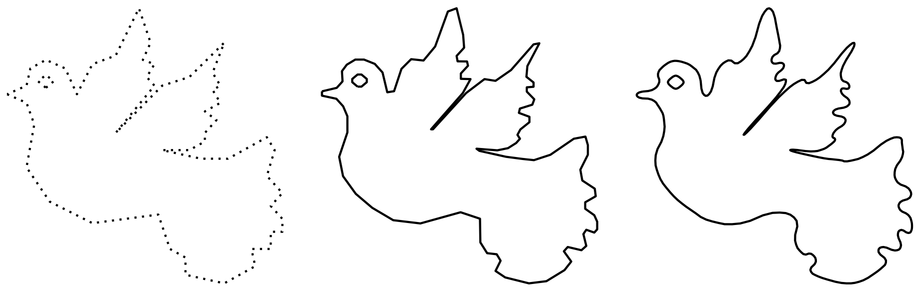


$(\mathbf{a})$ Initial polygon $(\mathbf{b})$ First level $(\mathbf{c})$ Third level

FIGURE 4. Present limit curves for open polygon produced by the scheme corresponding to A_3^4. The top portions of (b) and (c) represent the graph of $\cos x$.

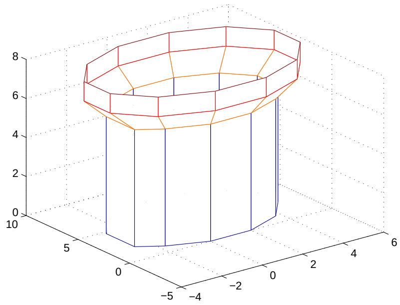


$(\mathbf{a})$ Initial polygon $(\mathbf{b})$ First level $(\mathbf{c})$ Third level

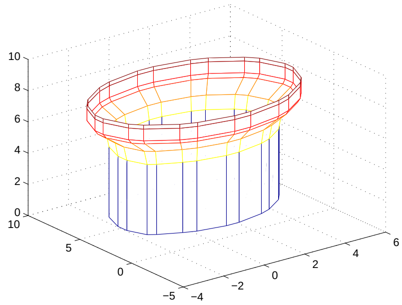


$(\mathbf{d})$ Initial polygon $(\mathbf{e})$ First level $(\mathbf{f})$ Third level

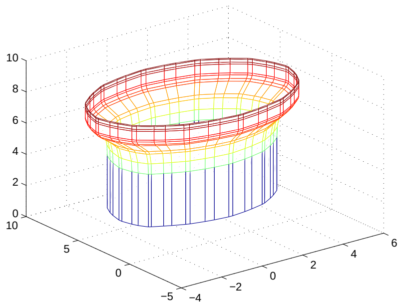
FIGURE 5. Present limit curves for close polygons produced by the scheme corresponding to A_3^2.



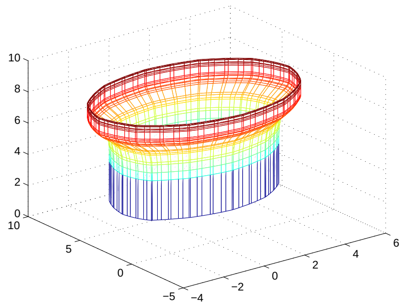
$(\mathbf{a})$ Initial mesh



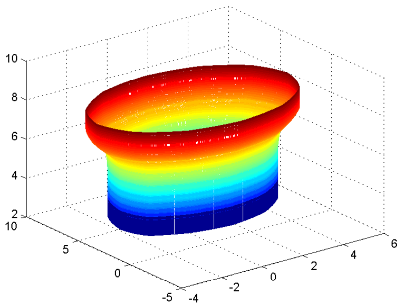
$(\mathbf{b})$ First level



$(\mathbf{c})$ Second level

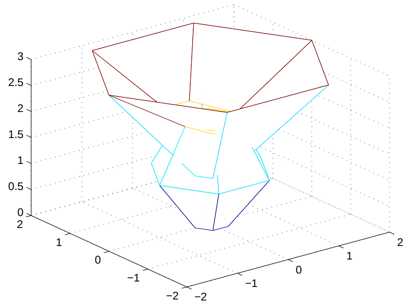


$(\mathbf{d})$ Third level

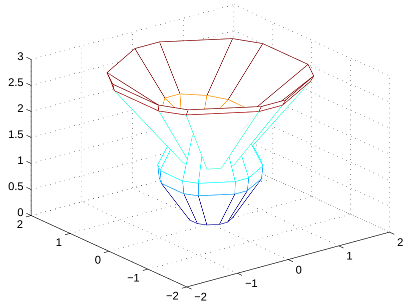


$(\mathbf{e})$ Limit surface

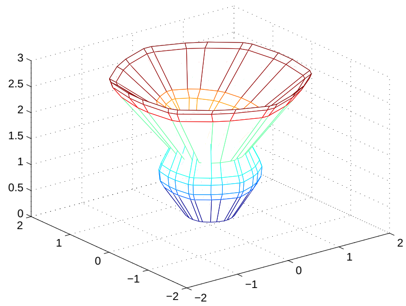
FIGURE 6. (a) Shows the initial mesh whereas (b)-(d) show the results after first, second and third subdivision levels (e) shows the limit surface produced by the 16-point tensor product binary scheme corresponding to A_5^2.



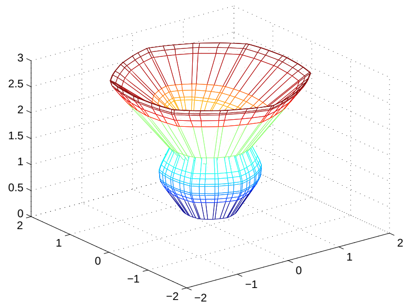
(**a**) Initial mesh



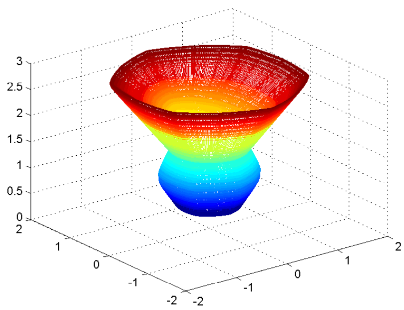
(**b**) First level



(**c**) Second level



(**d**) Third level



(**e**) Limit surface

FIGURE 7. (a) Shows the initial mesh whereas (b)-(d) show the results after first, second and third subdivision levels (e) shows the limit surface produced by the 16-point tensor product binary scheme corresponding to A_5^2.

7. CONCLUSION

In this paper, we have presented generalized symbols for univariate stationary and non-stationary subdivision schemes. In fact, our purposed algorithms are the extension of the well-known Lane-Riesenfeld algorithm in both stationary and non-stationary context. We can generate many families of any arity subdivision schemes. In particular, we have shown that uniform B-splines schemes, Hormann and Sabin [12] family of subdivision schemes and Novara and Romani [18] family of schemes are special cases of proposed algorithms. We also present the analysis of higher arity family of stationary subdivision schemes.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interest regarding the publication of this article and regarding the funding that they have received.

ACKNOWLEDGEMENT

This work is supported by National Research Programme for Universities (NRPU) (P. No. 3183) HEC, Pakistan. We thanks the anonymous reviewers for their careful reading of our manuscript and their many insightful comments and suggestions.

AUTHORS CONTRIBUTION

All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

REFERENCES

[1] P. Ashraf, G. Mustafa and J. Deng, *A six-point variant on the Lane-Riesenfeld algorithm*, Journal of Applied Mathematics, Article ID 628285, 07 pages, 2014.

[2] G. Chaikin, *An algorithm for high speed curve generation*, Computer Graphics and Image Processing **3**, (1974) 346-349.

[3] M. Charina, C. Conti and L. Romani, *Reproduction of exponential polynomials by multivariate non-stationary subdivision schemes with a general dilation matrix*, Numerische Mathematik, **127**, No. 2 (2014) 223-254.

[4] M. Charina, C. Conti, N. Guglielmi and V. Protasov, *Non-stationary multivariate subdivision: joint spectral radius and asymptotic similarity*, Mathematisches Forschungsinstitut Oberwolfach gGmbH **20**, 2013.

[5] C. Conti and L. Romani, *Affine combination of B-spline subdivision masks and its non-stationary counterparts*, BIT Numerical Mathematics **50**, No. 2 (2010) 269-299.

[6] C. Conti and K. Hormann, *Polynomial reproduction for univariate subdivision schemes of any arity*, Journal of Approximation Theory **163**, No. 4 (2011) 413-437.

[7] C. Conti, N. Dyn, C. Manni and M.-L. Mazure, *Convergence of univariate non-stationary subdivision schemes via asymptotic similarity*, Computer Aided Geometric Design **37**, (2015) 1-8.

[8] C. Conti and L. Romani, *Algebraic conditions on non-stationary subdivision symbols for exponential polynomial reproduction*, Journal of Computational and Applied Mathematics **236**, No. 4 (2011) 543-556.

[9] G. Deslauriers and S. Dubuc, *Symmetric iterative interpolation processes*, Constructive Approximation, **5**, No. 1 (1989) 49-68. 1989.

[10] N. Dyn and D. Levin, *Subdivision schemes in geometric modelling*, Acta Numerica **11**, (2002) 73 -144.

[11] M. S. Floater and G. Muntingh, *Exact regularity of symmetric univariate subdivision schemes", Geometry Seminar*, Centre of Mathematics for Applications, University of Oslow, Oslo, Norway, 2012.

[12] K. Hormann and M. A. Sabin, *A family of subdivision schemes with cubic precision*, Computer Aided Geometric Design **25**, No. 1 (2008) 41-52.

[13] I. P. Ivrissimtzis, M. A. Sabin and N. A. Dodgson, *On the support of recursive subdivision*, ACM Transactions on Graphics **23**, No. 4 (2004) 1043-1060.

[14] J. M. Lane and R. F. Riesenfeld, *A theoretical development for the computer generation and display of piecewise polynomial surfaces*, IEEE Transactions on Pattern Analysis and Machine Intelligence **2**, No. 1 (1980) 35-46.

[15] G. Mustafa, M. Bari and A. Sadiq, *The* $(2n)^2$*-point scheme based on bivariate cubic polynomial*, Punjab Univ. j. math. **49**, No. 3 (2017) 27-35.

[16] G. Mustafa and R. Hameed, *Families of univariate and bivariate subdivision schemes originated from quartic B-spline*, Advances in Computational Mathematics, (2017) 1-31.

[17] G. Mustafa and M. Bari, *Wide-ranging families of subdivision schemes for fitting data*, Punjab Univ. j. math. **48**, No. 2 (2016) 125-134.

[18] P. Novara and L. Romani, *Building blocks for designing arbitrarily smooth subdivision schemes with conic precision*, Journal of Computational and Applied Mathematics **279**, (2015) 67-79.

[19] O. Rioul, *Simple regularity criteria for subdivision schemes*, SIAM Journal on Mathematical Analysis **23**, No. 6 (1992) 1544-1576.