Punjab University
Journal of Mathematics (ISSN 1016-2526)
Vol. 48(2)(2016) pp. 125-134

Wide-Ranging Families of Subdivision Schemes for Fitting Data

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

Mehwish Bari
Department of Mathematics,
The Islamia University of Bahawalpur, Bahawalpur, Pakistan
Email: mehwishbari@yahoo.com

Received: 25 June, 2016 / Accepted: 22 August, 2016 / Published online: 05 September, 2016

Abstract. In this paper, we present wide-ranging families of subdivision schemes for fitting data to subdivision models. These schemes are constructed by fitting multivariate polynomial functions of any degree to different types of data by least squares techniques. Moreover, we also present the closed analytic expressions of the families of schemes for fitting data in 2 and 3 dimensional spaces. The schemes for fitting 3D data are non-tensor product schemes. Furthermore, it is straightforward by using our framework to construct schemes for fitting data in higher dimensional spaces. The performance of such schemes is demonstrated on examples of curves and surfaces.

AMS (MOS) Subject Classification Codes: 65D10; 65D17; 68W25; 93E24
Key Words: Subdivision schemes; Approximation; Least squares; Non-tensor product schemes; Approximating schemes.

1. Introduction

One of the most significant topics in the physical sciences is fitting data to subdivision models. The subdivision method is computational methods of linear and nonlinear fitting of smooth curves and surfaces to data. In addition, subdivision methods for fitting different types of data are still a current research topic in computer science. Subdivision scheme is an algorithm to generate smooth curve and surface as a sequence of successively refine polygonal mesh. This algorithm has sparse system of matrices [9] therefore problems can be handle easily.

A univariate a-ary subdivision scheme S which maps the coarse polygon f^k=$\{f_i^k\}_{i\in\mathbb{Z}}$ to refine polygon f^{k+1}=$\{f_i^{k+1}\}_{i\in\mathbb{Z}}$ at next refinement level is defined by

$$f_{ai+\mu}^{k+1} = \sum_{j\in\mathbb{Z}} \gamma_{aj+\mu} f_{i-j}^k, \quad \mu = 0, 1, ..., a-1,$$

where $\{..., \gamma_{i-1}, \gamma_i, \gamma_{i+1}, ...\}$ is called the mask and a is the arity (number of points inserted between two consecutive points) of scheme. The above rule can also be expressed as $f^{k+1} = Sf^k$. A necessary condition for the uniform convergence [3] of a-ary subdivision scheme is

$$\sum_{j \in \mathbb{Z}} \gamma_{aj+\mu} = 1, \quad \mu = 0, 1, ..., a-1.$$

Schemes are different due to their mask (i.e. the values of $\gamma_{aj+\mu}$), complexity (i.e. variation of j) and arity (i.e. the value of a: $a = 2, 3, ..,$ stands for binary, ternary and so on). The concept of computation of the mask by polynomial interpolation has been initiated by Deslauriers and Dubuc [5] in 1989. They have presented even-point a-ary schemes. After that different approaches were introduced to compute mask. Very recently, Dyn et al. [7] have presented univariate binary subdivision schemes whose mask is computed based on least squares minimization. But they have only presented closed analytic expressions of the univariate binary schemes. A recent paper [8] computes refined values by local l_1 optimization rather than by local least squares. They also presented closed analytic expressions of the univariate and bivariate binary schemes. These schemes are locally supported. Locally supported approximate identities on the unit ball have also been discussed by Akram et al. [1]. Recently, in 2015, Aslam [2] discussed the continuity of the schemes.
In this paper, we present closed analytic expressions of the families of $2n$- and $(2n+1)$-point a-ary approximating schemes for fitting 2D data. Moreover, we also present closed analytic expressions of the families of $4n^2$- and $(2n+1)^2$-point non-tensor product binary approximating schemes for fitting 3D data. These schemes are constructed by fitting univariate and bivariate polynomial functions of degree three to different types of data by least squares techniques. It is unambiguous to find schemes for fitting higher dimensional data by using our framework.

2. Univariate case: family of a-ary schemes

We first consider the univariate polynomial function of degree 3 to determine the best function to fit the data based on least squares then we construct $2n$- and $(2n+1)$-point a-ary schemes. A polynomial function of degree 3 is

$$f(x_r) = \eta_0 f_0(x_r) + \eta_1 f_1(x_r) + \eta_2 f_2(x_r) + \eta_3 f_3(x_r), \tag{2. 1}$$

where the monomials are defined as

$$f_0(x_r) = 1, \ f_1(x_r) = x_r, \ f_2(x_r) = x_r^2, \ f_3(x_r) = x_r^3.$$

The polynomial function (2. 1) with respect to the observations $(x_r = r, f_r)$ for $r = -n+1, \dots n$, and $n > 3$ can be written as

$$f_r = f(r) = \eta_0 + \eta_1 r + \eta_2 r^2 + \eta_3 r^3. \tag{2. 2}$$

Now determine the values of unknown parameters η's in (2. 2) to make the sum of squares of residuals as minimum. A residual has been defined as the difference between the observed value and the corresponding value of the function, that is

$$R = \sum_{r=-n+1}^{n} [f_r - \eta_0 - \eta_1 r - \eta_2 r^2 - \eta_3 r^3]^2. \tag{2. 3}$$

Differentiating R with respect to η's, setting each of the four equations to 0 and after solving the equations, we get the values of η_0, η_1, η_2 and η_3. By substituting $r = \frac{2q+1}{2a}$ in (2. 2) and then putting the values of η's in (2. 2), by changing notations, we get $2n$-point

a-ary approximating iterative scheme based on fitting polynomial function of degree 3 with monomial basis:

$$f_{ai+q}^{k+1} = \frac{1}{\phi_n} \sum_{r=-n+1}^{n} p_{r,n} f_r^k, \quad q = 0, 1, ..., a-1, \qquad (2.\ 4)$$

where $\phi_n = 8n\left(n^2-1\right)\left(4n^2-1\right)\left(4n^2-9\right)a^3$,

$$\begin{aligned} p_{r,n} \; = \;& 280\left\{\left(3n^2-3\right)a^3+\omega\left(-6n^2+11\right)a^2-15\omega^2 a+10\omega^3\right\}r^3+\left\{\left(-240n^4-\right.\right.\\ & \left.480n^2+720\right)a^3+600\omega\left(3n^2-5\right)a^2+360\omega^2\left(2n^2+13\right)a-\left.4200\omega^3\right\}r^2\\ & +\left\{\left(-360n^4+960n^2-600\right)a^3+200\omega\left(6n^4-18n^2+11\right)a^2+600\omega^2 a\right.\\ & \left.\left(3n^2-5\right)-280\omega^3\left(6n^2-11\right)\right\}r+24\left\{\left(6n^4-5n^2-6\right)a^3-5\omega\left(3n^2-\right.\right.\\ & \left.\left.5\right)a^2-10\omega^2\left(n^2+3\right)a+35\omega^3\right\}\left(n^2-1\right) \end{aligned}$$

and $\omega = q + \frac{1}{2}$.

2.1. **Alternatives and variants.** Thus far in this brief, we have focused our concentration to introduce scheme based on fitting of a polynomial function of degree 3 to $2n$ observations in two-dimensional space by least squares procedure. A further alternative can be obtain by fitting polynomial function of degree 3 to $2n+1$ observations. For example, for a slight variation on the scheme (2. 4), we suggest the replacement of $r = \frac{2q+1}{2a}$ by $r = \frac{q}{2a}$, $q = \pm1, \pm3, \pm5, ..., \pm(a-1)$ and again by $r = \frac{q}{2a}$, $q = 0, \pm1, \pm3, \pm5, ..., \pm(a-1)$ respectively in (2. 2), summation from $r = -n+1...n$ by $r = -n...n$ in (2. 3) and adopting the same procedure as above, we get following $(2n+1)$-point *even* a-ary approximating subdivision scheme

$$f_{ai+j}^{k+1} = \frac{1}{\psi_n} \sum_{r=-n}^{n} b_{r,n} f_r^k, \quad j = 0, 1, ..., (a-1) \qquad (2.\ 5)$$

and $(2n+1)$-point *odd* a-ary approximating subdivision scheme

$$f_{ai+j}^{k+1} = \frac{1}{\psi_n} \sum_{r=-n}^{n} b_{r,n} f_r^k, \quad j = 0, 1, ..., (a-1), \qquad (2.\ 6)$$

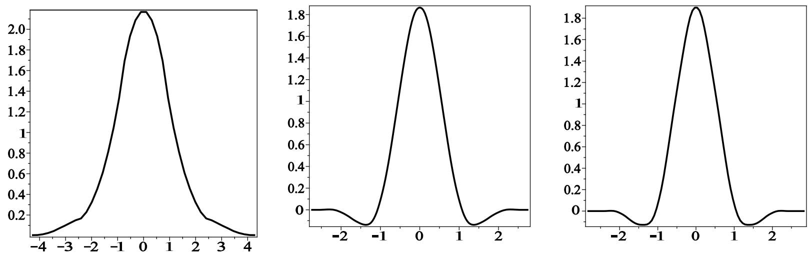
where $\psi_n = 8n\left(n+2\right)\left(2n+3\right)\left(n^2-1\right)\left(4n^2-1\right)a^3$ and

$$\begin{aligned} b_{r,n} = & -35q\left(12a^2n^2+12a^2n-4a^2-5q^2\right)r^3+30a\left(n-1\right)\left(n+2\right)\left(4a^2n^2+\right.\\ & \left.4a^2n-3q^2\right)r^2+5q\left(60a^2n^4+120a^2n^3-21q^2n^2-21nq^2-60a^2n+\right.\\ & \left.7q^2+20a^2\right)r+6na\left(n+2\right)\left(n^2-1\right)\left(12a^2n^2+12a^2n-4a^2-5q^2\right). \end{aligned}$$

A further generalization to yield schemes of the least squares procedure can be made by fitting polynomial function of degree less or greater than three. It is to be noted that the existing even-point approximating schemes are special cases of the scheme (2. 4) e.g. for $a = 2$ and $n = 2$, in (2. 4), we get 4-point binary approximating scheme of Dyn et al. [6].

It is amazing and astonishing that by changing monomial basis with Gram's, Laguerre, Legendre, Chebyshev and Hermite polynomials respectivley in (2. 1) and by adopting the same procedure, we get exactly the same schemes defined in (2. 4)-(2. 6).

2.2. **Basic limit functions.** Since for each $q = 0, 1, ..., a-1$, $\sum_{r=-n+1}^{n}\left(\frac{1}{\phi_n}\right)p_{r,n} = 1$ and for each $q = 0, \pm1, \pm3, \pm5, ..., \pm(a-1)$, $\sum_{r=-n}^{n}\left(\frac{1}{\psi_n}\right)b_{r,n} = 1$ so the basic conditions for the schemes, defined in (2. 4)-(2. 6), to be convergent are satisfied. Since every



(a) Binary (b) Ternary (c) Quaternary

FIGURE 1. *(a)-(c) show the basic limit functions of proposed 4-point binary, ternary and quaternary subdivision schemes.*

convergent subdivision scheme S is associated with a basic limit function (BLF), defined as $\phi_S = S^\infty f_i^0$ where f_i^0 be the initial data such that $f_i^0 = 1$, for $i = 0$ otherwise $f_i^0 = 0$, then it is necessary to compute the support width of $2n$- and $(2n+1)$-point a-ary approximating schemes. Generally the support of the scheme is also the support of its basic limit function and vice versa. The following prepositions can be easily proved by using the approach of Beccari et al. [4].

Theorem 2.3. *The BLF φ_{2n}^a of $2n$-point a-ary schemes have the support width $\kappa = \frac{2an-1}{a-1}$, $n \geq 2$ which implies that it vanishes outside the interval $\left[-\frac{2an-1}{2(a-1)}, \frac{2an-1}{2(a-1)}\right]$.*

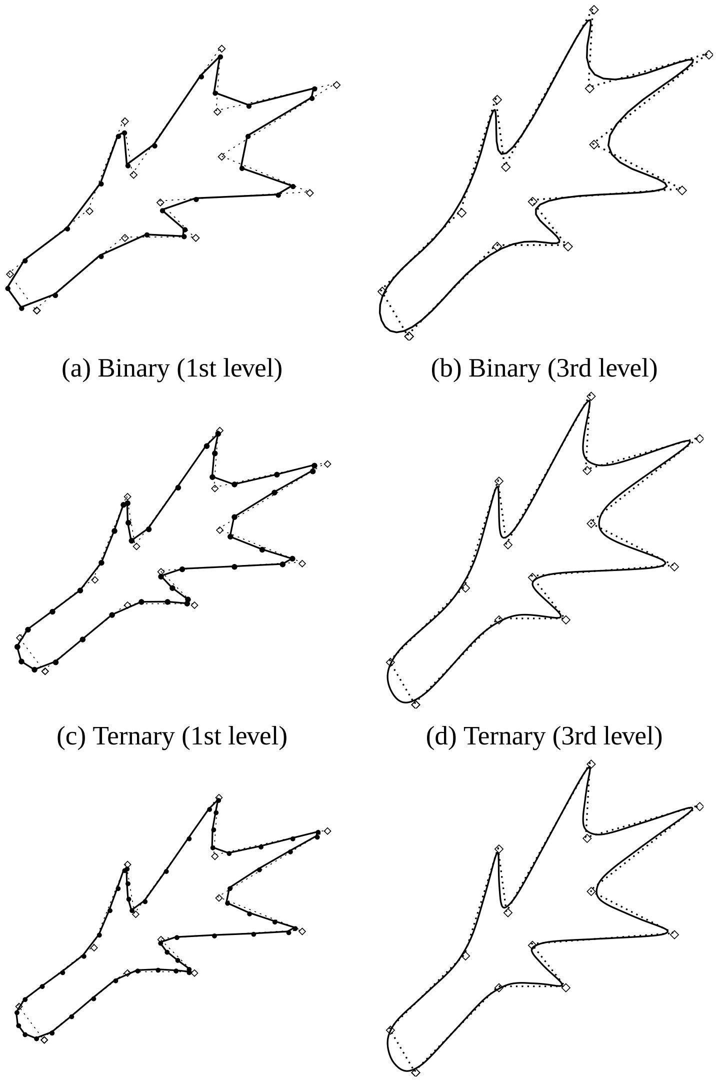
Theorem 2.4. *The BLF φ_{2n+1}^a of $(2n+1)$-point a-ary schemes have the support width $\kappa = \frac{(2n+1)a-1}{a-1}$, which implies that it vanishes outside the interval $\left[-\frac{(2n+1)a-1}{2(a-1)}, \frac{(2n+1)a-1}{2(a-1)}\right]$.*

It is observed from above prepositions that arity and support width are inversely proportional to each other. The BLF of proposed 4-point binary, ternary and quaternary schemes are depicted in Figure 1(a), 1(b) and 1(c) respectively.

2.5. **Comparison and performance.** The performance of different arity schemes is shown in Figures 2 and 3. Figure 2(a)-(c) show the results of binary, ternary and quaternary schemes at first iteration. This figure also show the functioning and operational behaviour of these schemes. Figure 2(d)-(f) show the limit curves shaped by binary, ternary and quaternary subdivision schemes respectively. The comparison of proposed schemes with the local least squares approach and B-spline scheme is shown in Figure 3. In this figure profile of car is produced by three different approaches. From this figure, we see that cubic polynomial generated by ordinary least squares is not fit for modeling car profile. It just shows the overall trend of the data/polygon. The proposed 4-point binary scheme generated by least squares based cubic polynomial is more consistent with the data/polygon comparative to B-spline.

3. BIVARIATE CASE: NON-TENSOR PRODUCT SCHEMES

In this section, we generalize our representation of previous section to the 3-dimensional case that is to construct a $4n^2$-point non-tensor product binary scheme based on least squares by fitting bivariate cubic polynomial function to data, we generalize the symmetric grid procedure i.e. $x_r = r$, $y_s = s$, $-n+1 \leqslant r, s \leqslant n$, $n > 1$. So a general bivariate



(a) Binary (1st level) (b) Binary (3rd level)

(c) Ternary (1st level) (d) Ternary (3rd level)

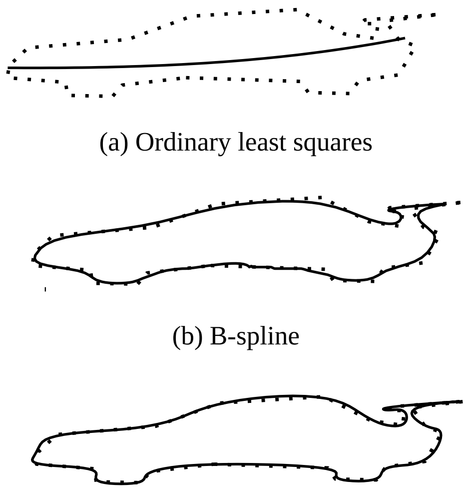
(e) Quaternary (1st level) (f) Quaternary (3rd level)

FIGURE 2. *(a), (c) and (e) show the behavior at first level/iteration while (b), (d) and (f) show the limit curves after third iteration of proposed 4-point binary, ternary and quaternary subdivision schemes respectively.*

polynomial function of degree three with respect to the observations $(x_r = r, y_s = s, f_{r,s})$ can be written as

$$f_{r,s} = f(r,s) = \beta_1 + \beta_2 r + \beta_3 s + \beta_4 r^2 + \beta_5 rs + \beta_6 s^2 + \beta_7 r^3 + \beta_8 r^2 s + \beta_9 rs^2 + \beta_{10} s^3.$$

Since the method of least squares calls for the selection of polynomial that minimizes R, the sum of the squares of differences between observed value $f_{r,s}$ and the corresponding



(a) Ordinary least squares

(b) B-spline

(c) 4-point binary scheme

FIGURE 3. *Dots show the initial polygon whereas (a) shows the least squares approach, (b) shows the behavior of B-spline and (c) shows the performance of proposed 4-point binary scheme.*

exact value $f(r,s)$. So by differentiating

$$R = \sum_{r=-n+1}^{n} \sum_{s=-n+1}^{n} [f_{r,s} - (\beta_1 + \beta_2 r + \beta_3 s + \beta_4 r^2 + \beta_5 rs + \beta_6 s^2 + \beta_7 r^3 + \beta_8 r^2 s + \beta_9 rs^2 + \beta_{10} s^3)]^2,$$

with respect to β_1, β_2,..., β_{10} and then setting them to 0, we get the ten normal equations. Solution of these equations gives the values of unknowns. By substituting these values in general bivariate polynomial function of degree three with variables x and y and then simplifying it, we get

$$f(x,y) = \frac{1}{\lambda}\left[\sum_{r=-n+1}^{n}\left\{\sum_{s=-n+1}^{n}\left(\beta_1 + \beta_2 x + \beta_3 y + \beta_4 x^2 + \beta_5 xy + \beta_6 y^2 + \beta_7 x^3 + \beta_8 x^2 y + \beta_9 xy^2 + \beta_{10} y^3\right) f_{r,s}\right\}\right], \quad (3.\ 7)$$

where

$\beta_1 = 9450r^3 + (1890s - 19845)\,r^2 + \left(1890s^2 - 3024s - 10773\right) r + 9450s^3 - 19845s^2 - 10773s + 35154,$

$\beta_2 = -13650r^3 + (1890s + 14805)\,r^2 - \left(3780s^2 - 378s - 31731\right) r + 1890s^2 - 3024s - 10773,$

$\beta_4 = -15750r^3 - (1890s - 29295)\,r^2 + (1890s + 14805)\,r + 1890s - 19845,$

$\beta_5 = (-3780s + 1890)\,r^2 - \left(3780s^2 - 10584s - 378\right) r + 1890s^2 + 378s - 3024,$

$\beta_7 = 1050\,(2r - 1)\left(5r^2 - 5r - 9\right),$

$\beta_8 = 1890\,(2s - 1)\left(r^2 - r - 1\right)$

and

$$\lambda = 4\left(n-1\right)\left(2n+3\right)\left(2n-3\right)\left(n+1\right)n^2\left(2n-1\right)^2\left(2n+1\right)^2. \qquad (3.\ 8)$$

By replacing r by s and s by r, we can get β_3 from β_2, β_6 from β_4, β_9 from β_8 and β_{10} from β_7.
Now by evaluating polynomial (3. 7) at particular points $(\frac{1}{4},\frac{1}{4})$, $(\frac{3}{4},\frac{1}{4})$, $(\frac{1}{4},\frac{3}{4})$, $(\frac{3}{4},\frac{3}{4})$, and then by changing notations we get $4n^2$-point non-tensor product binary approximating scheme with four rules: $f(\frac{1}{4},\frac{1}{4}) = f^k_{2i,2j}$, $f(\frac{3}{4},\frac{1}{4}) = f^k_{2i+1,2j}$, $f(\frac{1}{4},\frac{3}{4}) = f^k_{2i,2j+1}$ and $f(\frac{3}{4},\frac{3}{4}) = f^k_{2i+1,2j+1}$. As an example for $n = 2$, we get 16-point non-tensor product binary scheme. The performance of this scheme is shown in Figure 4(a)-4(e).

3.1. **Alternatives and variants.** A further slight variant of $4n^2$-point scheme can be made based on fitting of bivariate polynomial function of degree three to $(2n+1)^2$ observations, i.e. $(x_r = r, y_s = s, f_{r,s})$ for $-n \leqslant r, s \leqslant n$, $n > 1$, by least squares procedure. By adopting the similar approach, we get

$$f\left(x,y\right) = \frac{1}{\varpi}\left[\sum_{r=-n}^{n}\left\{\sum_{s=-n}^{n}\left(\beta_1+\beta_2 x+\beta_3 y+\beta_4 x^2+\beta_5 xy+\beta_6 y^2+\beta_7 x^3+\right.\right.\right.$$
$$\left.\left.\left.\beta_8 x^2 y+\beta_9 xy^2+\beta_{10} y^3\right)f_{r,s}\right\}\right] \qquad (3.\ 9)$$

where

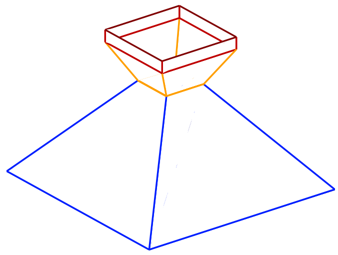
$$\begin{aligned}
&\beta_1 = \left(n-1\right)\left(n+2\right)n^2\left(n+1\right)^2\left(14n^2+14n-15r^2-15s^2-3\right),\\
&\beta_2 = 5rn\left(n+1\right)\left\{18n^4+36n^3-(9s^2+21r^2+3)n^2-(21r^2+9s^2+21)n\right.\\
&\qquad \left.+7r^2+18s^2+5\right\},\\
&\beta_4 = -15n\left(n-1\right)\left(n+1\right)\left(n+2\right)\left(n^2+n-3r^2\right),\\
&\beta_5 = 9rs\left(n-1\right)\left(n+2\right)\left(2n-1\right)\left(2n+3\right),\\
&\beta_7 = -35rn\left(n+1\right)\left(3n^2+3n-5r^2-1\right),\\
&\beta_8 = -45s\left(n-1\right)\left(n+2\right)\left(n^2+n-3r^2\right)
\end{aligned}$$

and

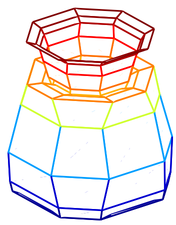
$$\varpi = \left(n-1\right)\left(n+2\right)\left(2n-1\right)\left(2n+3\right)n^2\left(n+1\right)^2\left(2n+1\right)^2.$$

By replacing r by s and s by r, we can get β_3 from β_2, β_6 from β_4, β_9 from β_8 and β_{10} from β_7.
Now by evaluating polynomial (3. 9) at particular points $(\frac{1}{4},\frac{1}{4})$, $(\frac{3}{4},\frac{1}{4})$, $(\frac{1}{4},\frac{3}{4})$, $(\frac{3}{4},\frac{3}{4})$, and then by changing notations we get $(2n+1)^2$-point non-tensor product binary approximating scheme. For example for $n = 2$, we get 25-point non-tensor product binary scheme. The presentation of this scheme is depicted in Figure 5.

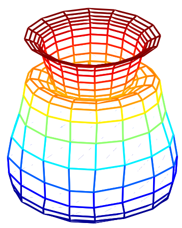
3.2. **Multivariate case: volumetric schemes.** A further generalization can be made based on fitting of a multivariate polynomial function of arbitrary degree to the observations. For example, one can get volumetric subdivision scheme for solid modelling by fitting trivariate polynomial function $f(x, y, z)$ to the observations $(x_r = r, y_s = s, z_t = t)$, for $-n+1 \leqslant r, s, t \leqslant n$ or $-n \leqslant r, s, t \leqslant n$, $t > 1$ by least squares method.



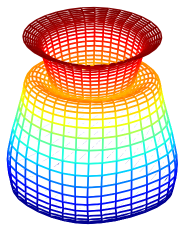
(a) Initial mesh



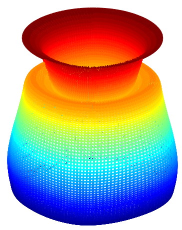
(b) First level



(c) Second level

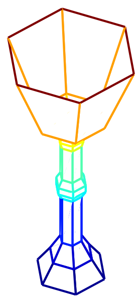


(d) Third level

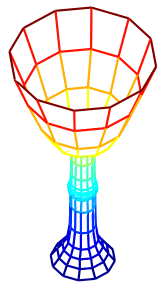


(e) Limit surface

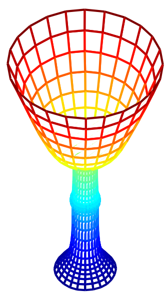
FIGURE 4. *(a) Shows the initial mesh whereas (b)-(d) show the results after first, second and third subdivision levels (e) shows limit surface produced by 16-point non-tensor product binary scheme.*



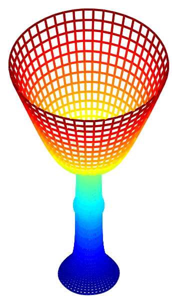
(a) Initial mesh



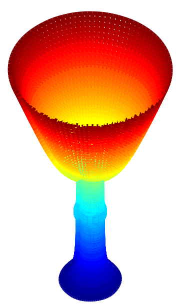
(b) First level



(c) Second level



(d) Third level



(e) Limit surface

FIGURE 5. *(a) Shows the initial mesh whereas (b)-(d) show the results after first, second and third subdivision levels (e) shows limit surface produced by 25-point non-tensor product binary scheme.*

3.3. **Conclusion.** The closed analytic expressions of the families of a-ary schemes for curve fitting have been presented. Moreover, the non-tensor product version of binary schemes with closed analytic form has been also introduced. Our proposed framework is easy to get variants and generalizations of these schemes by fitting univariate, bivariate and multivariate polynomial functions of any degree. The visual performances of these schemes have been checked by numerical examples.

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 NRPU Project No. 3183 and the Indigenous Ph. D Scholarship Scheme of Higher Education Commission (HEC) Pakistan.

REFERENCES

[1] M. Akram and V. Michel, *Locally supported approximate identities on the unit ball, Revista Matematica Complutense*, **23**, (2010) 233-249.

[2] M. Aslam, C^1 *continuity of 3-point nonlinear ternary interpolating subdivision schemes,* International Journal of Numerical Methods and Applications, **14**, No. 2 (2015) 119-132.

[3] N. Aspert, *Non-linear subdivision of univariate signals and discrete surfaces*, EPFL thesis, 2003.

[4] C. Beccari, G. Casiola and L. Romani, *An interpolating 4-point* C^2 *ternary non-stationary subdivision scheme with tension control*, Computer Aided Geometric Design, **24**, No. 4 (2007) 210-219.

[5] G. Deslauriers and S. Dubuc, *Symmetric iterative interpolation processes,* Constructive Approximation, **5**, (1989) 49-68.

[6] N. Dyn, M. S. Floater and K. A. Hormann, C^2 *four point subdivision scheme with fourth order accuracy and its extensions,* In: Mathematical Methods for Curves and Surfaces: Tromso 2004, M. Daehlen, K. Morken and L. L. Schumaker (eds.), 145-156, 2005.

[7] N. Dyn, A. Head, K. Hormann and N. Sharon, *Univariate subdivision schemes for noisy data with geometric applications,* Computer Aided Geomertric Design, **37**, (2015) 85-104.

[8] G. Mustafa, H. Li, J. Zhang and J. Deng, l_1*-regression based subdivision scheme for noisy data,* Computer Aided Design, **58**, (2015) 189-199.

[9] L. Xu, R. Wang, J. Zhang, Z. Yang, J. Deng, F. Chen and L. Liu, *Survey on sparsity in geometric modeling and processing,* **82**, (2015) 160-180.