*Open Journal of Applied Sciences*, **2013, 3, 263-269**
doi:10.4236/ojapps.2013.33033 Published Online July 2013 (http://www.scirp.org/journal/ojapps)

# A Unified Interpolating Subdivision Scheme for Curves/Surfaces by Using Newton Interpolating Polynomial

**Faheem Khan*, Irem Mukhtar, Noreen Batool**
Department of Mathematics, University of Sargodha, Sargodha, Pakistan
Email: *fahimscholar@gmail.com, irammukhtar1133@hotmail.com, noreen_batool@yahoo.com

Received April 1, 2013; revised May 12, 2013; accepted May 20, 2013



## ABSTRACT

This paper presents a general formula for (2*m* + 2)-point *n*-ary interpolating subdivision scheme for curves for any integer $m \geq 0$ and $n \geq 2$ by using Newton interpolating polynomial. As a consequence, the proposed work is extended for surface case, which is equivalent to the tensor product of above proposed curve case. These formulas merge several notorious curve/surface schemes. Furthermore, visual performance of the subdivision schemes is also presented.

**Keywords:** Interpolating Subdivision Scheme; Tensor Product Scheme; Newton Interpolating Polynomial

## 1. Introduction

Subdivision schemes have become important in recent years because of giving a specific and proficient way to describe smooth curve/surfaces. It is an algorithm method to generate smooth curve/surfaces as a sequence of successively refined polyhedral meshes. Their beauty lies in the elegant mathematical formulation and simple implementation. The flexibility and simplicity of subdivision schemes become more appropriate in computer and industrial applications.

There are two general classes of subdivision scheme namely interpolating and approximating. If the limit curve/surface approximates the initial control polygon and that subdivision, the newly generated control points are not in the limit curve/surface, the scheme is said to be approximating. It is called interpolating if after subdivision, the control points are interpolated on the limit curve/surface. Among interpolating subdivision scheme 4-point interpolating scheme [1] was one of the initial scheme. Nowadays spacious mixture of interpolating scheme [2-8] has been anticipated in the literature with different shape parameters.

In 1978, Catmull-Clark [9] and Doo-Sabin [10] first introduced subdivision surface schemes, which generalised the tensor product of bicubic and biquadratic B-splines respectively. After that, Kobbelt [11] gave the tensor product of the curve case and he generalized the four-point interpolatory subdivision scheme for curve to the surface by using tensor product.

The proposed work gives a new idea in finding subdivision rules for curves and surfaces using Newton interpolating polynomial. The proposed method is simple and avoids complex computation when deriving subdivision rules. Since higher arity subdivision schemes have high approximation order and lower support than their counterpart of lower arity schemes. Therefore researchers are focusing in introducing higher arity schemes (*i.e.*, ternary, quaternary, ..., *n*-ary). This paper presents a general formula for (2*m* + 2)-point *n*-ary interpolating rules for curves. Since the subdivision schemes for surface design have gained more popularity in computer animation industries. So, a new approach for regular quad meshes using 2-dimensional Newton interpolating formula is also the part of this paper.

In the following section, there is presented a brief introduction about the preliminary concepts used in this work. In Sections 3 and 4, new formula for interpolating subdivision schemes is given for curves and surfaces by using Newton interpolating polynomial. In Section 5, application of the subdivision schemes is also accessible. A few remarks and conclusions are given in Section 6.

## 2. Preliminaries

Given a sequence of control points $p_i^k \in \mathbb{R}^N, i \in \mathbb{Z}, N \geq 1,$ where the upper index $k \geq 0$

*Corresponding author.

indicates the subdivision level. An $n$-ary subdivision curve is defined by

$$p_{ni+\alpha}^{k+1}=\sum_{j=0}^{m}a_{\alpha,j}\,p_{i+j}^{k},\quad \alpha=0,1,\dots,n-1,\qquad(2.1)$$

where $m > 0$ and

$$\sum_{j=0}^{m}a_{\alpha,j}=1,\quad \alpha=0,1,\dots,n-1.\qquad(2.2)$$

The set of coefficients $\left\{a_{\alpha,\mathrm{j}},\alpha=0,1,\dots,n-1\right\}_{j=0}^{m}$ is called subdivision mask. In the limit $k\to\infty$, the process (2.1) defines an infinite set of points in $\mathbb{R}^N$. The sequence of control points $\left\{p_i^k\right\}$ is connected, in a natural way, with the diadic mesh points $t_i^k=i/n^k, i\in\mathbb{Z}$. The process then defines a scheme whereby $p_{ni}^{k+1}$ and $p_{ni+n}^{k+1}$ replaces the values $p_i^k$ and $p_{i+1}^k$ at the mesh points S and $t_{ni+n}^{k+1}=t_{i+1}^k$ respectively, $p_{ni+\alpha}^{k+1}$ while is inserted at the new mesh points

$$t_{ni+\alpha}^{k+1}=\frac{1}{n}\left((n-\alpha)t_i^k+\alpha t_{i+1}^k\right)\ \text{ for }\ \alpha=1,2,\dots,n-1.$$

Labeling of old and new points is shown in **Figure 1,** which illustrates subdivision scheme (2.1).

Let $\prod_{2m+1}$ be the space of all polynomials of degree $\le 2m+1$, where $m$ is non-negative integer. If $\left\{\mathrm{N}_j(x)\right\}_{j=-m}^{m+1}$ is fundamental Newton polynomial corresponding to the node point $\{j\}_{j=-m}^{m+1}$ is defined by

$$P_{2m+1}(x)=\sum_{j=-m}^{m+1}a_jN_j(x)\qquad(2.3)$$

In general, the coefficient of the Newton form of polynomial is called divided difference, the divided difference $a_j=y[\mathrm{x}_0,\dots,\mathrm{x}_n]$, is a symmetric function, hence can be found by following method,

$$a_j=p[x_{-m},\dots,x_j]=\begin{cases}y_{-m}, & x_m=x_j\\ \displaystyle\sum_{i=-m}^{j}\frac{p_i}{\prod_{i=-m,i\ne k}^{j}(i-k)}, & x_m\ne x_j,\end{cases}\qquad(2.4)$$

and $\mathrm{N}_j(\mathrm{x})$ can originate by the subsequent way,

$$N_j(x)=\begin{cases}1, & k=j\\ \displaystyle\prod_{k=-m}^{j-1}(i-k), & k=j,\end{cases}\qquad(2.5)$$

## 3. Construction of the Subdivision Scheme for Univariate Case

This section gives the construction of $(2m + 2)$-point binary and ternary interpolating schemes. Then by induction, a general formula for $(2m + 2)$-point $n$-ary interpolating subdivision scheme is formulated for curve case.

### 3.1. (2*m* + 2)-Point Binary Interpolating Scheme

To construct the rules for binary 2-point interpolating scheme, consider $\left\{\mathrm{N}_j(x)\right\}_{j=0}^{1}$ be the Fundamental Newton polynomial to the node points $\{0, 1\}$. The Newton polynomial replicate linear polynomial $P$ in the way that taking $m = 0$ in (2.3), we achieve

$$P_1(x)=\sum_{j=0}^{1}a_jN_j(x),\qquad(3.1)$$

where $a_j$ is divided difference can be calculated by (2.4), and $\mathrm{N}_j(\mathrm{x})$ by setting in (2.5). This implies that

$$\begin{aligned}P_1(x)&=p_0+(p_1-p_0)(x)\\&=\sum_{\mu=0}^{1}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu}C_{\nu}^{\mu}p_{\nu}\right]C_{\mu}^{x}(\Gamma),\end{aligned}\qquad(3.2)$$

with following Gamma function

$$C_{\mu}^{x}(\Gamma)=\frac{\Gamma(x+1)}{\Gamma(x+1-\mu)\Gamma(\mu+1)}.$$

Now, to construct the desired 2-point ternary subdivision scheme, let

$$p_{2i}^{1}=p_1(i+0),\qquad p_{2i+1}^{1}=p_1\left(i+\frac{1}{2}\right).$$

Since we want to construct uniform and stationary scheme reproducing polynomials up to a fixed degree, it is sufficient to consider the case $i = 0$ with subdivision level $k = 0$. This implies that

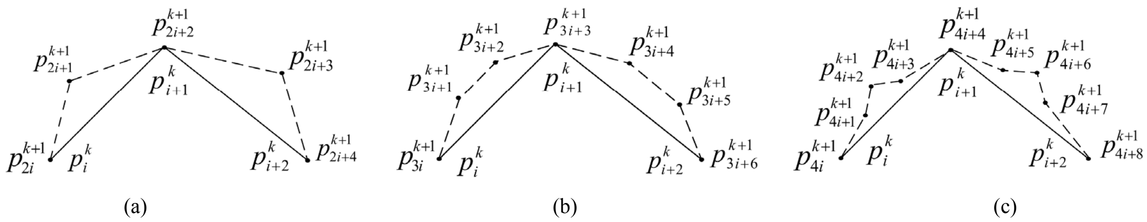

**Figure 1. Solid lines show coarse polygons whereas dotted lines are refined polygons. (a)-(c) represent binary, ternary and quaternary refinement of coarse polygon using (2.1) for n = 2, 3, 4 respectively.**

$$p_1(0) = p_0,$$

$$p_1\left(\frac{1}{2}\right) = \frac{1}{2}p_0 + \frac{1}{2}p_1.$$

Now as an affine combination of 2-point $p_i^{k+1}$, $p_{i+1}^{k+1}$, we suppose that at $(k+1)$th level, the point $p_{2i+\alpha}^{k+1}$ is attached to the parametric value $\frac{2i+\alpha}{2^{k+1}}$, so the desired binary 2-point interpolating subdivision scheme is given by

$$\begin{cases} p_{2i}^{k+1} = p_i^k, \\ p_{2i+1}^{k+1} = \frac{1}{2}p_i^k + \frac{1}{2}p_{i+1}^k. \end{cases} \tag{3.3}$$

In composite form (3.3) can be written as

$$p_{2i+\alpha}^{k+1} = \sum_{\mu=0}^{1}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{\nu}^{\mu} p_{i+\nu}\right] C_{\mu}^{x}(\Gamma), \quad \alpha = 0,1 \tag{3.4}$$

where

$$\{p_i^k\} C_{\mu}^{x}(\Gamma) = \frac{\Gamma(x+1)}{\Gamma(x+1-\mu)\Gamma(\mu+1)}, \qquad x = \frac{\alpha}{2}.$$

Continuing in the same way for $m = 1$ in (2.3), where $\mathrm{N}_j(x)$ be the Newton polynomial to the node points $\{-1, 0, 1, 2\}$ then we have the following compact form of 4-point binary subdivision scheme

$$p_{2i+\alpha}^{k+1} = \sum_{\mu=0}^{3}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{\nu}^{\mu} p_{i+\nu-1}\right] C_{\mu}^{x+1}(\Gamma), \quad \alpha = 0,1 \tag{3.5}$$

where

$$C_{\mu}^{x+1}(\Gamma) = \frac{\Gamma(x+2)}{\Gamma(x+2-\mu)\Gamma(\mu+1)}, \qquad x = \frac{\alpha}{2}.$$

Consequently, we can generate a general form of $(2m + 2)$-point binary interpolating scheme, which is of the following form

$$p_{2i+\alpha}^{k+1} = \sum_{\mu=0}^{2m+1}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{i+\nu}^{\mu} p_{i+(\nu-m)}\right] C_{\mu}^{x+m}(\Gamma), \tag{3.6}$$

where

$$C_{\mu}^{x+m}(\Gamma) = \frac{\Gamma(x+m+1)}{\Gamma(x+m+1-\mu)\Gamma(\mu+1)}, \ \alpha = 0,1$$

corresponding to $x = \frac{\alpha}{2}$, $m \geq 0$ and subdivision level $k \geq 0$.

### 3.2. (2*m* + 2)-Point Ternary Interpolating Scheme

To construct the rules for ternary 4-point interpolating scheme, consider $\{\mathrm{N}_j(x)\}_{j=-1}^{2}$ be the Newton polynomial to the node points $\{-1, 0, 1, 2\}$. The Newton polynomial reproduces cubic polynomial $P$ in the way that taking $m = 1$ in (2.3), we achieve

$$P_3(x) = \sum_{j=-1}^{2} a_j N_j(x). \tag{3.7}$$

Now by using (2.4) and (2.5) in (3.5), we have

$$\begin{aligned} P_3(x) &= p_{-1} + (p_0 - p_{-1})(x+1) \\ &+ \frac{1}{2}(p_1 - 2p_0 + p_{-1})(x^2 + x) \\ &+ \frac{1}{6}(p_2 - 3p_1 + 3p_0 - p_{-1})(x^3 - x) \\ &= \sum_{\mu=0}^{3}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{i+\nu}^{\mu} p_{\nu-1}\right] C_{\mu}^{x+1}(\Gamma). \end{aligned} \tag{3.8}$$

Now to construct the 4-point ternary subdivision scheme, take

$$p_{3i}^1 = p_3(i+0), \quad p_{3i+1}^1 = p_3\left(i+\frac{1}{3}\right), \quad p_{3i+2}^1 = p_3\left(i+\frac{2}{3}\right). \tag{3.9}$$

From (3.7) and (3.8), we get

$$p_3(0) = p_0,$$

$$p_3\left(\frac{1}{3}\right) = -\frac{5}{81}p_{-1} + \frac{20}{27}p_0 + \frac{10}{27}\mathrm{p}_1 - \frac{4}{80}p_2,$$

$$p_3\left(\frac{2}{3}\right) = -\frac{4}{81}p_{-1} + \frac{10}{27}p_0 + \frac{20}{27}\mathrm{p}_1 - \frac{5}{80}p_2,$$

we attain the following iterative rules for ternary 4-point interpolating subdivision scheme,

$$\begin{cases} p_{3i}^{k+1} = p_i^k, \\ p_{3i+1}^{k+1} = -\frac{5}{81}p_{i-1}^k + \frac{20}{27}p_i^k + \frac{10}{27}p_{i+1}^k - \frac{4}{80}p_{i+2}^k, \\ p_{3i+2}^{k+1} = -\frac{4}{81}p_{i-1}^k + \frac{10}{27}p_i^k + \frac{20}{27}p_{i+1}^k - \frac{5}{80}p_{i+2}^k. \end{cases} \tag{3.10}$$

In composite form, the above rules can be written as

$$p_{3i+\alpha}^{k+1} = \sum_{\mu=0}^{3}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{\nu}^{\mu} p_{i-(\nu-1)}^k\right] C_{\mu}^{x+1}(\Gamma), \alpha = 0,1,2 \tag{3.11}$$

where

$$C_{\mu}^{x}(\Gamma) = \frac{\Gamma(x+2)}{\Gamma(x+2-\mu)\Gamma(\mu+1)}, \qquad x = \frac{\alpha}{3}.$$

Accordingly, general formula for $(2m + 2)$-point ternary interpolating scheme is given by

$$p_{3i+\alpha}^{k+1} = \sum_{\mu=0}^{2m+1}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu} C_{\nu}^{\mu} p_{i+(\nu-m)}\right] C_{\mu}^{x+m}(\Gamma), \tag{3.12}$$

where

$$C_{\mu}^{x+m}(\Gamma)=\frac{\Gamma(x+m+1)}{\Gamma(x+m+1-\mu)\Gamma(\mu+1)},\ \alpha=0,1,2$$

corresponding to $x=\frac{\alpha}{3}$, $m\geq 0$ and subdivision level $k\geq 0$.

### 3.3. (2*m* + 2)-Point *n*-Ary Interpolating Subdivision Scheme (Generalization)

Now there is presented a general formula for (2*m* + 2)-point *n*-ary (*i.e.* binary, ternary and so on) interpolating subdivision scheme by using Newton interpolating polynomial. This new formula will be helpful to drive interpolating subdivision rule plainly and quickly. The general formula for (2*m* + 2)-point *n*-ary interpolating subdivision scheme has the following form

$$p_{ni+\alpha}^{k+1}=\sum_{\mu=0}^{2m+1}(-1)^{\mu}\left[\sum_{\nu=0}^{\mu}(-1)^{\nu}C_{\nu}^{\mu}p_{i+(\nu-m)}\right]C_{\mu}^{x+m}(\Gamma),\quad(3.13)$$

where

$$C_{\mu}^{x+m}(\Gamma)=\frac{\Gamma(x+m+1)}{\Gamma(x+m+1-\mu)\Gamma(\mu+1)},\ \alpha=0,1,\ldots,n-1$$

corresponding to $x=\frac{\alpha}{n}$, $m\geq 0$ and $k\geq 0$ indicates the subdivision level, where *n* stand for *n*-ary scheme.

**Remark 3.1.** In the following, we see that some other well-known interpolating schemes come from our proposed Formula (3.13).

- Setting the value of $m$ = 1, and $n$ = 2, in above result, which is the 4-point DD scheme [12],

$$\begin{cases}p_{2i}^{k+1}=p_i^k,\\ p_{2i+1}^{k+1}=-\frac{1}{16}p_{i-1}^k+\frac{9}{16}p_i^k+\frac{9}{16}p_{i+1}^k-\frac{1}{16}p_{i+2}^k.\end{cases}$$

- By setting $m$ = 2, and $n$ = 2, in proposed result, we get 6-point DD scheme [12],

$$\begin{cases}p_{2i}^{k+1}=p_i^k,\\ p_{2i+1}^{k+1}=\frac{3}{256}p_{i-2}^k-\frac{25}{256}p_{i-1}^k+\frac{75}{128}p_i^k\\ \qquad+\frac{75}{128}p_{i+1}^k-\frac{25}{256}p_{i+2}^k+\frac{3}{256}p_{i+3}^k.\end{cases}$$

- Taking $m$ = 1, and $n$ = 3 in (3.13), we get ternary 4-point interpolating scheme [13],

$$\begin{cases}p_{2i}^{k+1}=p_i^k,\\ p_{3i+1}^{k+1}=-\frac{5}{81}p_{i-1}^k+\frac{20}{27}p_i^k+\frac{20}{27}p_{i+1}^k-\frac{4}{81}p_{i+2}^k,\\ p_{3i+2}^{k+1}=-\frac{4}{81}p_{i-1}^k+\frac{20}{27}p_i^k+\frac{20}{27}p_{i+1}^k-\frac{5}{81}p_{i+2}^k\end{cases}$$

- By setting $m$ = 2, and $n$ = 3 in (3.13), we get ternary 6-point interpolating scheme [13],

$$\begin{cases}p_{2i}^{k+1}=p_i^k,\\ p_{3i+1}^{k+1}=-\frac{5}{81}p_{i-1}^k+\frac{20}{27}p_i^k+\frac{20}{27}p_{i+1}^k-\frac{4}{81}p_{i+2}^k,\\ p_{3i+2}^{k+1}=-\frac{4}{81}p_{i-1}^k+\frac{20}{27}p_i^k+\frac{20}{27}p_{i+1}^k-\frac{5}{81}p_{i+2}^k\end{cases}$$

## 4. Tensor Product of (2*m* + 2)-Point *n*-Ary Interpolating Subdivision Scheme

Given a sequence of control points $p_{i,j}^k\in\mathbb{R}^N$, $i,j\in\mathbb{Z}$, $N\geq 2$, where the upper index $k\geq 0$ indicates the subdivision level. An *n*-ary subdivision surface scheme in the tensor product form is defined by

$$p_{ni+\alpha,nj+\beta}^{k+1}=\sum_{r=0}^{m}\sum_{s=0}^{m}a_{\alpha,r}a_{\beta,s}p_{i+r,j+s}^k,\ \alpha,\beta=0,1,\ldots,n-1,\quad(4.1)$$

where $\{a_{\alpha,r}\}_{r=0}^m$ and $\{a_{\beta,s}\}_{s=0}^m$ satisfy (2.2). Given initial values $p_{i,j}^0\in\mathbb{R}^N$, $i,j\in\mathbb{Z}$, then in the limit $k\to\infty$ the process (4.1) defines an infinite set of points in $\mathbb{R}^N$. The sequence of values $\{p_{i,j}^k\}$ is related, in a natural way, with the diadic mesh points $\left(\frac{i}{n^k},\frac{j}{n^k}\right)$, $i,j\in\mathbb{Z}$. The process then defines a scheme whereby $p_{ni+\alpha,nj+\beta}^{k+1}$ replaces the value $p_{i+\alpha/n,j+\beta/n}^k$ at the mesh point $\left(\frac{i+\alpha/n}{n^k},\frac{j+\beta/n}{n^k}\right)$ for $\alpha,\beta\in(0,\mathrm{n})$, while the values $p_{ni+\alpha,nj+\beta}^{k+1}$ are inserted at the new mesh points $\left(\frac{ni+\alpha}{n^{k+1}},\frac{nj+\beta}{n^{k+1}}\right)$ for $\alpha,\beta\in 0,1,\ldots,n-1$. Labeling of old and new points is shown in **Figure 2** which illustrates subdivision scheme (4.1).

Here, we present a general formula for tensor product of (2*m*+2)-point *n*-ary interpolating scheme in the following form,

$$p_{n_1i+\alpha,n_2j+\beta}^{k+1}=\sum_{\mu_1=0}^{2m_1+1}\sum_{\mu_2=0}^{2m_2+1}(-1)^{\mu_1+\mu_2}\left[\sum_{\nu_1=0}^{\mu_1}\sum_{\nu_2=0}^{\mu_2}(-1)^{\nu_1+\nu_2}\right.$$
$$\left.C_{\nu_1}^{\mu_1}C_{\nu_2}^{\mu_2}p_{i+(\nu_1-m_1),j+(\nu_2-m_2)}\right]\times C_{\mu_1}^{x_1+m_1}(\Gamma)C_{\mu_2}^{x_2+m_2}(\Gamma),\quad(4.2)$$

where

$$C_{\mu_1}^{x_1+m_1}(\Gamma)=\frac{\Gamma(x_1+m_1+1)}{\Gamma(x_1+m_1+1-\mu_1)\Gamma(\mu_1+1)},$$
$$C_{\mu_2}^{x_2+m_2}(\Gamma)=\frac{\Gamma(x_2+m_2+1)}{\Gamma(x_2+m_2+1-\mu_2)\Gamma(\mu_2+1)}.$$

ss

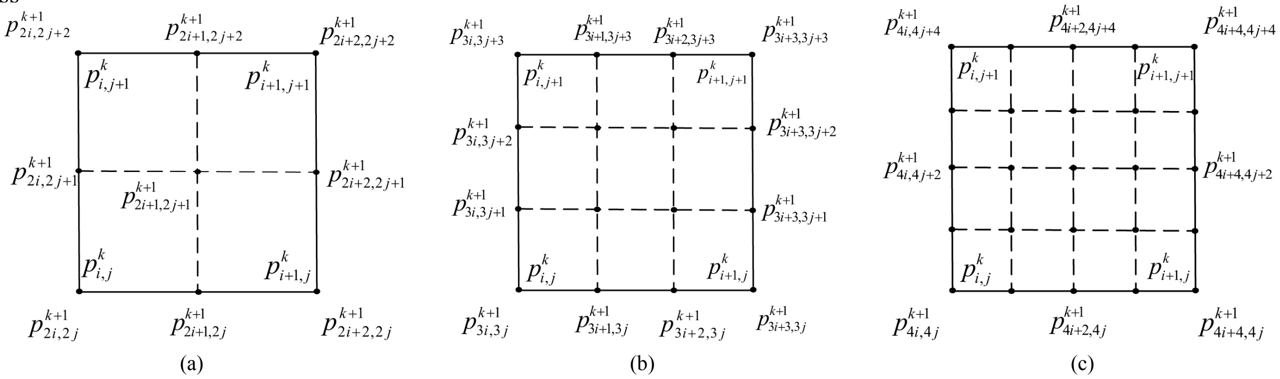


**Figure 2. Solid lines show one face of coarse polygons whereas dotted lines are refined polygons. (a)-(c) can be obtained by subdividing one face into four, nine and sixteen new faces by using (4.1) for n = 2, 3, 4 respectively.**

Here, $\alpha = 0,1,...,n_1 - 1$ corresponding to $x = \dfrac{\alpha}{n_1}$, $\beta = 0,1,...,n_2 - 1$ corresponding to $x = \dfrac{\beta}{n_2}$, $m \geq 0$, $k \geq 0$ indicate the subdivision level and $n_1, n_2 \geq 2$.

**Remark 4.1.** In the following, it is to be noted that some of the bivariate interpolating subdivision schemes come from our proposed formula (4.2).

- For obtaining Kobbelt [11] subdivision scheme, substitute $n_1, n_2 = 2$ and $m_1, m_2 = 2$ in (4.2), we have the following refinement rules,

$$
\begin{cases}
p^{k+1}_{2i,2j} = p^k_{i,j}, \\
p^{k+1}_{2i,2j+1} = -\dfrac{1}{16} p^k_{i-1,j} + \dfrac{9}{16} p^k_{i,j} + \dfrac{9}{16} p^k_{i+1,j} - \dfrac{1}{16} p^k_{i+2,j}, \\
p^{k+1}_{2i+1,2j} = -\dfrac{1}{16} p^k_{i,j-1} + \dfrac{9}{16} p^k_{i,j} + \dfrac{9}{16} p^k_{i,j+1} - \dfrac{1}{16} p^k_{i,j+2}, \\
p^{k+1}_{2i+1,2j+1} = \dfrac{1}{256} p^k_{i-1,j-1} - \dfrac{9}{256} p^k_{i-1,j} - \dfrac{9}{256} p^k_{i-1,j+1} \\
\qquad + \dfrac{1}{256} p^k_{i-1,j+2} - \dfrac{9}{256} p^k_{i,j-1} + \dfrac{81}{256} p^k_{i,j} \\
\qquad + \dfrac{81}{256} p^k_{i,j+1} - \dfrac{9}{256} p^k_{i,j+2} - \dfrac{9}{256} p^k_{i+1,j-1} \\
\qquad + \dfrac{81}{256} p^k_{i+1,j} + \dfrac{81}{256} p^k_{i+1,j+1} - \dfrac{9}{256} p^k_{i+1,j+2} \\
\qquad + \dfrac{1}{256} p^k_{i+2,j-1} - \dfrac{9}{256} p^k_{i+2,j} \\
\qquad - \dfrac{9}{256} p^k_{i+2,j+1} + \dfrac{1}{256} p^k_{i+2,j+2}.
\end{cases}
$$

- For obtaining tensor product of ternary 4-point interpolating scheme, taking the values $n_1, n_2 = 3$ and $m_1, m_2 = 1$ in (4.2), we get the following refinement rules,

$$p^{k+1}_{3i,3j} = p^k_{i,j},$$

$$p^{k+1}_{3i,3j+1} = -\frac{5}{81} p^k_{i,j-1} + \frac{20}{27} p^k_{i,j} + \frac{10}{27} p^k_{i,j+1} - \frac{4}{81} p^k_{i,j+2},$$

$$p^{k+1}_{3i,3j+2} = -\frac{4}{81} p^k_{i,j-1} + \frac{10}{27} p^k_{i,j} + \frac{20}{27} p^k_{i,j+1} - \frac{5}{81} p^k_{i,j+2},$$

$$p^{k+1}_{3i+1,3j} = -\frac{5}{81} p^k_{i-1,j} + \frac{20}{27} p^k_{i,j} + \frac{10}{27} p^k_{i=1,j} - \frac{4}{81} p^k_{i+2,j},$$

$$
\begin{aligned}
p^{k+1}_{3i+1,3j+1} = {} & \frac{25}{6561} p^k_{i-1,j-1} - \frac{100}{2187} p^k_{i-1,j} - \frac{50}{2187} p^k_{i-1,j+1} \\
& + \frac{20}{6561} p^k_{i-1,j+2} - \frac{100}{2187} p^k_{i,j-1} + \frac{400}{729} p^k_{i,j} \\
& + \frac{200}{729} p^k_{i,j+1} - \frac{80}{2187} p^k_{i,j+2} - \frac{50}{2187} p^k_{i+1,j-1} \\
& + \frac{200}{729} p^k_{i+1,j} + \frac{100}{729} p^k_{i+1,j+1} - \frac{40}{2187} p^k_{i+1,j+2} \\
& + \frac{20}{6561} p^k_{i+2,j-1} - \frac{80}{2187} p^k_{i+2,j} \\
& - \frac{40}{2187} p^k_{i+2,j+1} + \frac{16}{6561} p^k_{i+2,j+2},
\end{aligned}
$$

$$
\begin{aligned}
p^{k+1}_{3i+1,3j+2} = {} & \frac{20}{6561} p^k_{i-1,j-1} - \frac{50}{2187} p^k_{i-1,j} - \frac{100}{2187} p^k_{i-1,j+1} \\
& + \frac{25}{6561} p^k_{i-1,j+2} - \frac{80}{2187} p^k_{i,j-1} + \frac{200}{729} p^k_{i,j} \\
& + \frac{400}{729} p^k_{i,j+1} - \frac{100}{2187} p^k_{i,j+2} - \frac{40}{2187} p^k_{i+1,j-1} \\
& + \frac{100}{729} p^k_{i+1,j} + \frac{200}{729} p^k_{i+1,j+1} - \frac{50}{2187} p^k_{i+1,j+2} \\
& + \frac{16}{6561} p^k_{i+2,j-1} - \frac{40}{2187} p^k_{i+2,j} \\
& - \frac{80}{2187} p^k_{i+2,j+1} + \frac{20}{6561} p^k_{i+2,j+2},
\end{aligned}
$$

$$p^{k+1}_{3i+2,3j} = -\frac{4}{81} p^k_{i-1,j} + \frac{10}{27} p^k_{i,j} + \frac{20}{27} p^k_{i=1,j} - \frac{5}{81} p^k_{i+2,j},$$

 

$$
\begin{aligned}
p^{k+1}_{3i+2,3j+1} = {} & \frac{20}{6561} p^k_{i-1,j-1} - \frac{80}{2187} p^k_{i-1,j} - \frac{40}{2187} p^k_{i-1,j+1} \\
& + \frac{16}{6561} p^k_{i-1,j+2} - \frac{50}{2187} p^k_{i,j-1} + \frac{200}{729} p^k_{i,j} \\
& + \frac{100}{729} p^k_{i,j+1} - \frac{40}{2187} p^k_{i,j+2} - \frac{100}{2187} p^k_{i+1,j-1} \\
& + \frac{400}{729} p^k_{i+1,j} + \frac{200}{729} p^k_{i+1,j+1} - \frac{80}{2187} p^k_{i+1,j+2} \\
& + \frac{25}{6561} p^k_{i+2,j-1} - \frac{100}{2187} p^k_{i+2,j} \\
& - \frac{50}{2187} p^k_{i+2,j+1} + \frac{20}{6561} p^k_{i+2,j+2},
\end{aligned}
$$

$$
\begin{aligned}
p^{k+1}_{3i+2,3j+2} = {} & \frac{16}{6561} p^k_{i-1,j-1} - \frac{40}{2187} p^k_{i-1,j} - \frac{80}{2187} p^k_{i-1,j+1} \\
& + \frac{20}{6561} p^k_{i-1,j+2} - \frac{40}{2187} p^k_{i,j-1} + \frac{100}{729} p^k_{i,j} \\
& + \frac{200}{729} p^k_{i,j+1} - \frac{50}{2187} p^k_{i,j+2} - \frac{80}{2187} p^k_{i+1,j-1} \\
& + \frac{200}{729} p^k_{i+1,j} + \frac{400}{729} p^k_{i+1,j+1} - \frac{100}{2187} p^k_{i+1,j+2} \\
& + \frac{20}{6561} p^k_{i+2,j-1} - \frac{50}{2187} p^k_{i+2,j} \\
& - \frac{100}{2187} p^k_{i+2,j+1} + \frac{25}{6561} p^k_{i+2,j+2}.
\end{aligned}
$$

**Lemma 4.1.** [14] Given initial control polygon $p^0_{i,j} = p_{i,j}$, $i, j \in \mathbb{Z}$, let the values $p^k_{i,j}, k \geq 1$ be defined recursively by subdivision process (4.1) together with (2.2), then the schemes derived by tensor product naturally get four-sided support regions.

**Remark 4.2.** It can be loosely say that the support is the tensor product of the supports of the two regions, just as one can loosely say that Kobbelt subdivision scheme for surface [11] is the generalization of the tensor product 4-point DD subdivision scheme [12].

**Lemma 4.2.** [15] Given initial control polygon $p^0_{i,j} = p_{i,j}$, $i, j \in \mathbb{Z}$, let the values $p^k_{i,j}, k \geq 1$ be defined recursively by subdivision process (4.1) together with (2.2), then if a scheme is derived from a tensor product, then the level of continuity can be determined between pieces by reference to the underlying basis functions, *i.e.*, all the tensor product schemes have the same continuity as their counterparts.

## 5. Application

This section is devoted for the visual performance of curves/surfaces. It is illustrated by some examples, obtained from the proposed work (3.13) and (4.2). The stepwise subdivision effects are shown in **Figures 3** and **4**.

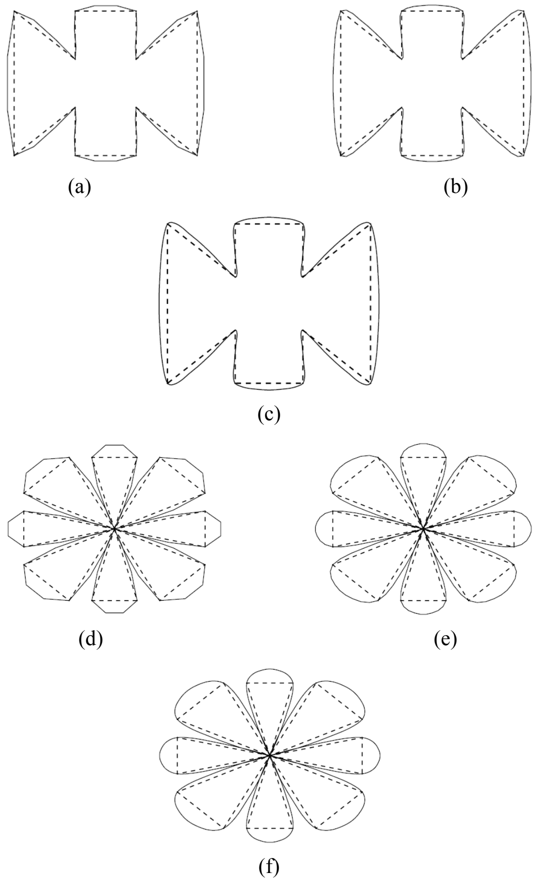


**Figure 3. Dotted line indicate initial polygon whereas continuous curve generated by ternary 4- and 6-point interpolating subdivision schemes [12]. (a) 4-point: 1st level; (b) 2nd level; (c) 3rd level; (d) 6-point: 1st level; (e) 2nd level; (f) 3rd level.**

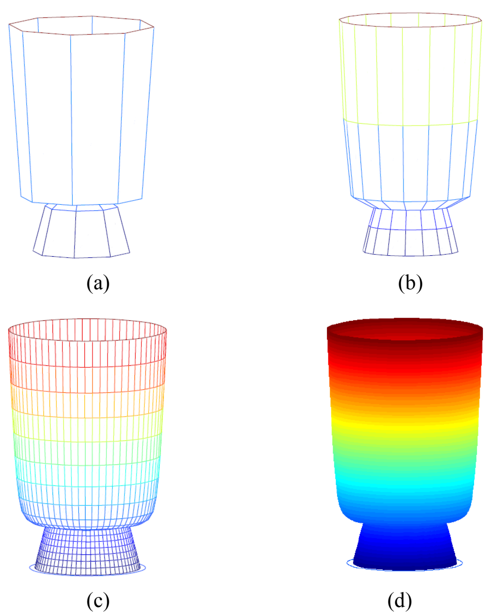


**Figure 4. Tensor product of 4-point binary approximating scheme: (a)-(d) show the initial polygon, 1st-, 2nd-subdivision levels and limit surface respectively.**

## 6. Conclusion

This work gives a variety of subdivision schemes for the univariate and bivariate cases by using Newton's interpolating formula. The work presented here is a new approach to the subdivision rules, which reduce the computational cost. Most of the well-known subdivision schemes are the special cases of the proposed work (3.13) and (4.2).

## 7. Acknowledgements

The authors are pleased to acknowledge the mysterious referees for their careful reading of the manuscript and precious comments, which made this manuscript more constructive. First author pays special thanks to Dr. Shahid Mubeen for his assistance in research.

## REFERENCES

[1] N. Dyn, D. Levin and J. A. Gregory, "A 4-Point Interpolatory Subdivision Scheme for Curve Design," *Computer Aided Geometric Design*, Vol. 24, No. 4, 1987, pp. 257-268. doi:10.1016/0167-8396(87)90001-X

[2] C. Beccari, G. Casciola and L. Romani, "An interpolating 4-Point $C^2$ Ternary Non-Stationary Subdivision Scheme with Tension Control," *Computer Aided Geometric Design*, Vol. 24, No. 4, 2007, pp. 210-219. doi:10.1016/j.cagd.2007.02.001

[3] N. Dyn, "Interpolatory Subdivision Scheme and Analysis of Convergence and Smoothness by the Foralism of Laurent Polynomials," In: A.Iske, E. Quak and M. S. Floater, Eds., *Tutorials on Multiresolution in Geometric Modelling*, Springer, Berlin, 2002, pp. 51-68 (Chapter 2 and 3).

[4] M. F. Hassan and N. A. Dodgson, "Ternary and Three-Point Univariate Subdivision Schemes," In: A. Cohen, J. L. Marrien and L. L. Schumaker, Eds., *Curve and Surface Fitting*: *Saint-Malo*, 2002, Nashboro Press, Brentwood, 2003, pp. 199-208.

[5] M. F. Hassan, I. P. Ivrissimitzis and N. A. Dodgson, "An Interpolating 4-Point $C^2$ Ternary Stationary Subdivision Scheme," *Computer Aided Geometric Design*, Vol. 19, No. 1, 2002, pp. 1-18. doi:10.1016/S0167-8396(01)00084-X

[6] M. K. Jena, P. Shunmugaraj and P. C. Das, "A Non-Stationary Subdivision Scheme for Curve Interpolation," *Anziam Journal*, Vol. 44, 2003, pp. 216-235.

[7] G. Mustafa and F. Khan, "Ternary Six-Point Interpolatig Subdivision Scheme," *Lobachevskii Journal of Mathematics*, Vol. 29, No. 3, 2008, pp. 153-163. doi:10.1134/S1995080208030062

[8] A. Weissman, "A 6-Point Interpolatory Subdivision Scheme for Curve Design," M.Sc. Thesis, Tel Aviv University, Tel Aviv, 1990.

[9] E. Catmull and J. Clark, "Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes," *Computer Aided Design*, Vol. 10, No. 6, 1978, pp. 350-355. doi:10.1016/0010-4485(78)90110-0

[10] Doo and M. Sabin, "Bahaviour of Recursive Division Durfaces near Extraordinary Points," *Computer Aided Design*, Vol. 10, No. 6, 1978, pp. 356-360. doi:10.1016/0010-4485(78)90111-2

[11] L. Kobbelt, "Interpolatory Subdivision on Open Quadrilateral Nets with Arbitrary Topology," Computer Graphics Forum, Vol. 15, No. 3, 1996, pp. 409-420. doi:10.1111/1467-8659.1530409

[12] G. Deslauries and S. Dubuc, "Symmetric Iterative Interpolation Process," *Constructive Approximation*, Vol. 5, No. 1, 1989, pp. 49-68. doi:10.1007/BF01889598

[13] J. A. Lian, "On A-Aray Subdivision for Curve Design: 4-Point and 6-Point Interpolatory Scheme," *Applications and Applied Mathematics*: *An International Journal*, Vol. 3, No. 1, 2008, pp. 18-29.

[14] M. Sabin, "Eigenanalysis and Artifacts of Subdivision Curves and Surfaces," In: A. Iske, E. Quak and M. S. Floater, Eds., *Tutorials on Multiresolution in Geometric Modelling*, Springer, Berlin, 2002, pp. 51-68 (Chapter 4).

[15] N. A. Dodgson, U. H. Augsdorfer, T. J. Cashman and M. A. Sabin, "Deriving Box-Spline Subdivision Schemes," Springer-Verlag, Berlin, LNCS-5654, 2009, pp. 106-123.

 