

Article

# A Novel Numerical Algorithm to Estimate the Subdivision Depth of Binary Subdivision Schemes

Aamir Shahzad <sup>1</sup>, Faheem Khan <sup>1</sup>, Abdul Ghaffar <sup>2</sup> , Ghulam Mustafa <sup>3</sup>,  
Kottakkaran Sooppy Nisar <sup>4,\*</sup>  and Dumitru Baleanu <sup>5,6,7</sup> 

<sup>1</sup> Department of Mathematics, University of Sargodha, Sargodha 40100, Pakistan; uos.amir@gmail.com (A.S.); fahimscholar@gmail.com (F.K.)

<sup>2</sup> Department of Mathematical Sciences, BUIITEMS, Quetta 87300, Pakistan; abdulghaffar.jaffar@gmail.com

<sup>3</sup> Department of Mathematics, The Islamia University of bahawalpur, Punjab 63100, Pakistan; ghulam.mustafa@iub.edu.pk

<sup>4</sup> Department of Mathematics, College of Arts and Sciences, Prince Sattam bin Abdulaziz University, Wadi Aldawaser 11991, Saudi Arabia

<sup>5</sup> Department of Mathematics, Cankaya University, Ankara 06530, Turkey; dimitru@cankaya.edu.tr or Baleanu@mail.cmuh.org.tw

<sup>6</sup> Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40447, Taiwan

<sup>7</sup> Institute of Space Sciences, 077125 Magurele-Bucharest, Romania

\* Correspondence: n.sooppy@psau.edu.sa or ksnisar1@gmail.com; Tel.: +966-563456976

Received: 5 December 2019; Accepted: 22 December 2019; Published: 1 January 2020



**Abstract:** Subdivision schemes are extensively used in scientific and practical applications to produce continuous geometrical shapes in an iterative manner. We construct a numerical algorithm to estimate subdivision depth between the limit curves/surfaces and their control polygons after  $k$ -fold subdivisions. In this paper, the proposed numerical algorithm for subdivision depths of binary subdivision curves and surfaces are obtained after some modification of the results given by Mustafa et al in 2006. This algorithm is very useful for implementation of the parametrization.

**Keywords:** numerical analysis; algorithm; subdivision schemes; error bounds; subdivision depth

**MSC:** 65D17; 65D05; 65U07

## 1. Introduction

Computer Aided Geometric Design (CAGD) foremost deals with curves/surfaces and its computational aspects. Subdivision schemes (SSs) have achieved much popularity in the past few years because of its competitive generation of curves/surfaces, and this magnificence lies in its implementation along with its mathematical formulation. Here we provide an overview of CAGD, subdivision and some other related concepts. The term CAGD was first suggested by Riesenfeld and Barnhill [1] in 1974 and the initial work was performed in the 1960's. "Computational Geometry for Design and Manufacture" was the first book on CAGD authored by Faux and Pratt [2] in 1979. Subdivision is the most remarkable field for the purpose of modeling in CAGD, and is described as "the act or process of dividing something into smaller parts". A small number of papers have been published in the field of SSs for error dominance.

Huawei et al. [3] performed error (distance) estimations for the Doo–Sabin scheme. After this, Mustafa et al. [4–7] estimated error bounds for binary, non-stationary binary, ternary and quaternary SSs. Mustafa et al. computed error bounds for tensor product volumetric models and  $\sqrt{3}$  subdivision surfaces [8,9]. They also computed triangular surfaces subdivision depths [10]. However, Mustafa and Hashmi [11] determined subdivision depth computation for  $n$ -ary subdivision curves/surfaces

by using the first forward difference technique. Moncayo and Amat [12] estimated error bounds for the class of SSs based on the two-scale refinement equation. The authors in [4–11] imposed these conditions to find their results. But, it has been observed that all SSs do not satisfy the imposed condition. Amat et al. [12] relaxed the condition in [4] for estimating the error bounds of binary SSs for curve designing. In this paper, we relax the conditions in the work of [5–11] and generalize the work of [12] to estimating error bounds and subdivision depths of binary SSs for curve and regular surface design.

However, as of yet, no one has found a numerical algorithm to estimate error bounds and subdivision depths of parametric SSs for curve and regular surface design. This motivates us to present a novel numerical algorithm for estimating the subdivision depth of binary subdivision schemes. This paper introduces not only a numerical algorithm for parametric SSs, but also the estimation for the non-parametric SSs. This novel technique has lesser computational cost than numerous other techniques.

Now we present some preliminaries. In the case of the curve: Let  $\{f_i^k; i \in \mathbf{Z}\}$  be a sequence of points in  $\mathbf{R}^N$ , where  $N \geq 2$  and  $k$  be the non-negative integer which indicates the SS level. A generalized univariate binary SS [13] is illustrated in the following as

$$\begin{cases} f_{2i}^{k+1} = \sum_{m=0}^t a_{0,m} f_{i+m}^k \\ f_{2i+1}^{k+1} = \sum_{m=0}^t a_{1,m} f_{i+m}^k \end{cases} \tag{1}$$

with necessary condition of the convergence

$$\sum_{m=0}^t a_{0,m} = \sum_{m=0}^t a_{1,m} = 1. \tag{2}$$

By [4], we have

$$\begin{aligned} b_{0,m} &= \sum_{l=0}^m (a_{0,l} - a_{1,l}), \\ b_{1,m} &= a_{0,m} - b_{0,m}, \end{aligned} \tag{3}$$

such that

$$\sum_{m=0}^t |b_{0,m}| < 1 \quad \text{and} \quad \sum_{m=0}^t |b_{1,m}| < 1.$$

We introduce the coefficients for  $m = 0, 1, \dots, t$ , such that

$$\begin{cases} d_{2m} = b_{0,m}, \\ d_{2m+1} = b_{1,m}. \end{cases} \tag{4}$$

In the case of the surface: The points  $\{f_{i,j}^k; i, j \in \mathbf{Z}\}$  represents a sequence in  $\mathbf{R}^N$ ,  $N \geq 2$  and is described as

$$\begin{cases} f_{2i,2j}^{k+1} = \sum_{p=0}^t \sum_{q=0}^t a_{0,p} a_{0,q} f_{i+p,j+q}^k \\ f_{2i,2j+1}^{k+1} = \sum_{p=0}^t \sum_{q=0}^t a_{0,p} a_{1,q} f_{i+p,j+q}^k \\ f_{2i+1,2j}^{k+1} = \sum_{p=0}^t \sum_{q=0}^t a_{1,p} a_{0,q} f_{i+p,j+q}^k \\ f_{2i+1,2j+1}^{k+1} = \sum_{p=0}^t \sum_{q=0}^t a_{1,p} a_{1,q} f_{i+p,j+q}^k \end{cases} \tag{5}$$

with the condition

$$\sum_{p=0}^t a_{0,p} = \sum_{p=0}^t a_{1,p} = \sum_{q=0}^t a_{0,q} = \sum_{q=0}^t a_{1,q} = 1. \tag{6}$$

By [4], we have

$$\begin{aligned} \sum_{p=0}^t |a_{0,p}| \sum_{q=0}^t |b_{0,q}| < 1, \quad \sum_{p=0}^t |a_{0,p}| \sum_{q=0}^t |b_{1,q}| < 1, \\ \sum_{p=0}^t |a_{1,p}| \sum_{q=0}^t |b_{0,q}| < 1, \quad \sum_{p=0}^t |a_{1,p}| \sum_{q=0}^t |b_{1,q}| < 1, \end{aligned}$$

where  $b_{0,q}$  and  $b_{1,q}$  for  $q = 0, 1, \dots, t$  are defined in (3). We introduce the coefficients  $(f_n)_{n \in \mathbb{N}}, (d_n)_{n \in \mathbb{N}}$  for  $r, s = 0, 1, \dots, t$ , such that

$$\begin{cases} f_{2r} = a_{0,t-r} \\ f_{2r+1} = a_{1,t-r} & r = 0, \dots, t. \\ d_{2s} = b_{0,t-s} \\ d_{2s+1} = b_{1,t-s}, & s = 0, \dots, t. \end{cases} \tag{7}$$

## 2. Preliminaries Results

Here we present some preliminary notations and results. Readers are referred to [12] for more details.

### 2.1. Univariate Case

Let  $(u_n)_{n \geq 0}$  be the finite length vector and  $(d_n)_{n \geq 0} = (d_n)_{n=0}^{2N-1}$  with  $d_n = 0$  for  $n \geq 2N$ . The one time convolution product of  $u = (u_n)_{n \geq 0}$  and  $d = (d_n)_{n \geq 0}$  for binary subdivision curves is given by

$$(u^{(0)} \star d)_j = \sum_{n=0}^{[j/2]} u_n d_{j-2n}. \tag{8}$$

Similarly, we have the following reformulation for  $k_0$  convolutions

$$((\dots(((u^{(0)} \star d)^{(0)} \star d)^{(0)} \star \dots \star d)^{(0)} \star d)_j = \sum_{m=0}^{[j/2^{k_0}]} u_m C_{m,j}^{[k_0]}, \tag{9}$$

with

$$\begin{cases} C_{m,j}^{[1]} = A_{m,j} = d_{j-2m}, \\ C_{m,j}^{[k_0]} = \sum_{n=2m}^{[j/2^{k_0-1}]} A_{m,n} C_{n,j}^{[k_0-1]}, & k_0 \geq 2. \end{cases} \tag{10}$$

Hence by (9), we get

$$\|((\dots(((u^{(0)} \star d)^{(0)} \star d)^{(0)} \star \dots \star d)^{(0)} \star d)\|_\infty \leq \|u\|_\infty \max_j \left\{ \sum_{m=0}^{[j/2^{k_0}]} |C_{m,j}^{[k_0]}| \right\}, \tag{11}$$

and

$$\sup_j \left\{ \sum_{m=0}^{[j/2^{k_0}]} |C_{m,j}^{[k_0]}| \right\} = \sup_{j \in \Sigma(k_0, N)} \left\{ \sum_{m=0}^{[j/2^{k_0}]} |C_{m,j}^{[k_0]}| \right\}, \tag{12}$$

where

$$\Sigma(k_0, N) = \{\Omega(k_0, N) - 2^{k_0} + 1, \Omega(k_0, N) - 2^{k_0} + 2, \dots, \Omega(k_0, N)\}, \tag{13}$$

and

$$\Omega(k_0, N) = (2^{k_0} - 1)(2N - 1). \tag{14}$$

**Lemma 1.** *The associated constant of  $k_0$ -th convolution with coefficients  $d = (d_0, d_1, \dots, d_{2N-1})$  for binary subdivision curves is defined as*

$$D_{k_0} = \sup_{j \in \Sigma(k_0, N)} \left\{ \sum_{m=0}^{[j/2^{k_0}]} |C_{m,j}^{[k_0]}| \right\}. \tag{15}$$

### 2.2. Bivariate Case

Let  $(u_n)_{n \geq 0}$  be a vector and  $(d_n)_{n \geq 0} = (d_n)_{n=0}^{2N-1}$ ,  $(f_n)_{n \geq 0} = (f_n)_{n=0}^{2N-1}$  with  $d_n = f_n = 0$  for  $n \geq 2N$ . The convolution product of  $u = (u_n)_{n \geq 0}$ ,  $d = (d_n)_{n \geq 0}$  and  $f = (f_n)_{n \geq 0}$  for binary subdivision surfaces is given by

$$u_{i,j}^{k_0} = \left( u^{k_0-1;0} * fd \right)_{i,j} = \sum_{m=0}^{[i/2]} \sum_{n=0}^{[j/2]} u_{m,n}^{k_0-1} d_{j-2n} f_{i-2m}. \tag{16}$$

Similarly, we get the following reformulation of  $k_0$  convolutions for surface case

$$u_{i,j}^{k_0} = (\dots(((u^{k_0-1;0} * fd) * fd) * \dots * fd) * fd)_{i,j} = \sum_{m=0}^{[i/2^{k_0}]} \sum_{n=0}^{[j/2^{k_0}]} u_{m,n}^0 C_{n,j}^{[k_0,d]} C_{m,i}^{[k_0,f]}, \tag{17}$$

with

$$\begin{cases} C_{n,j}^{[k_0,d]} = \sum_{s=2n}^{[j/2^{k_0-1}]} C_{n,s}^{[k_0-1,d]} C_{s,j}^{[k_0-1,d]}, \\ C_{m,i}^{[k_0,f]} = \sum_{p=2m}^{[i/2^{k_0-1}]} C_{m,p}^{[k_0-1,f]} C_{p,i}^{[k_0-1,f]}. \end{cases} \tag{18}$$

From (17), we have

$$\max_{i,j} |u_{i,j}^{k_0}| \leq \max_{m,n} |u_{m,n}^0| \max_{i,j} \sum_{m=0}^{[i/2^{k_0}]} \sum_{n=0}^{[j/2^{k_0}]} |C_{n,j}^{[k_0,d]}| |C_{m,i}^{[k_0,f]}|, \tag{19}$$

and

$$\max_{i,j} \left\{ \sum_{m=0}^{[i/2^{k_0}]} \sum_{n=0}^{[j/2^{k_0}]} |C_{n,j}^{[k_0,d]}| |C_{m,i}^{[k_0,f]}| \right\} = \max_{i,j \in \Sigma(k_0, N)} \left\{ \sum_{m=0}^{[i/2^{k_0}]} \sum_{n=0}^{[j/2^{k_0}]} |C_{n,j}^{[k_0,d]}| |C_{m,i}^{[k_0,f]}| \right\}, \tag{20}$$

where  $\Sigma(k_0, N)$  is defined in (13).

**Lemma 2.** *The associated constants of  $k_0$ -th convolution with coefficients  $d = (d_0, d_1, \dots, d_{2N-1})$  and  $f = (f_0, f_1, \dots, f_{2N-1})$  for binary subdivision surfaces are defined as*

$$F_{k_0} = \sup_{i \in \Sigma(k_0, N)} \left\{ \sum_{m=0}^{[i/2^{k_0}]} |C_{m,i}^{[k_0,f]}| \right\}, \tag{21}$$

$$G_{k_0} = \sup_{j \in \Sigma(k_0, N)} \left\{ \sum_{n=0}^{[j/2^{k_0}]} |C_{n,j}^{[k_0,d]}| \right\}. \tag{22}$$

### 3. Subdivision Depth for Binary Subdivision Curves

Now firstly, the modified technique for finding the error bounds is presented. Secondly, an altered subdivision depth computation technique based on the proposed error bounds is demonstrated with the help of tables.

**Theorem 1.** Consider  $f_i^0, i \in \mathbb{Z}$  to be the control polygon and  $f_i^k$  to be the values for non-negative integers recursively described by (1) together with mask condition (2). Suppose  $F^k$  denotes the initial at the points  $\{f_i^k\}$ . Then after two successive iterations the error bounds between  $k$  and  $k + 1$  stage is

$$\|F^{k+1} - F^k\|_\infty \leq \gamma\beta D_{k_0}, \tag{23}$$

where  $D_{k_0}, k_0 \geq 1$  defined in (15) and  $\beta = \max_i \|\Delta p_i^0\|$ ,

$$\begin{aligned} \gamma &= \max \left( \sum_{j=0}^{t-1} |\tilde{a}_{0,j}|, \sum_{j=0}^{t-1} |\tilde{a}_{1,j}| \right), \quad \tilde{a}_{0,j} = \sum_{i=j+1}^t a_{0,i}, \\ \tilde{a}_{1,0} &= \sum_{i=1}^t a_{1,i} - \frac{1}{2}, \quad \tilde{a}_{1,j} = \sum_{i=j+1}^t a_{1,i}, \quad j \neq 0. \end{aligned}$$

**Theorem 2.** Under the same conditions used in Theorem 1, let  $F^\infty$  be the limit curve associated with the subdivision process, then

$$\|F^\infty - F^k\|_\infty \leq \gamma\beta \left( \frac{D_{k_0}^k}{1 - D_{k_0}} \right), \tag{24}$$

where  $k_0 \geq 1$  be a natural number, such that  $D_{k_0} < 1$ .

**Remark 1.** In this paper  $D_{k_0}$  for  $k_0 = 1$  is equal to  $\delta$  defined in [4]. Note that in [4], if  $\delta > 1$  then the error bounds can not be computed. But by [12] and by using our technique, we increase the value of  $k_0$  until  $D_{k_0}$  becomes less than one.

**Theorem 3.** Let  $k$  be the subdivision depth and let  $\nabla^k$  be the error bound between binary subdivision curve  $F^\infty$  and its  $k$ -level control polygon  $F^k$ . For arbitrary  $\epsilon > 0$ , if

$$k \geq \log_{D_{k_0}^{-1}} \left( \frac{\gamma\chi}{\epsilon(1 - D_{k_0})} \right), \tag{25}$$

then  $\nabla^k \leq \epsilon$ .

**Proof.** Let  $\nabla^k$  be the distance between limit curve  $F^\infty$  and control polygon  $F^k$  at the  $k$ -th subdivision level, then

$$\nabla^k = \|F^\infty - F^k\|_\infty \leq \gamma\beta \left( \frac{D_{k_0}^k}{1 - D_{k_0}} \right). \tag{26}$$

To obtain the given error tolerance  $\epsilon > 0$ , the subdivision depth  $k$  satisfies the following relation, which is in inequality form as

$$k \geq \log_{D_{k_0}^{-1}} \left( \frac{\gamma\chi}{\epsilon(1 - D_{k_0})} \right), \tag{27}$$

then  $\nabla^k \leq \epsilon$ .

Hence, this completes the proof.  $\square$

Application for Univariate Case

Here, we present some numerical examples to compute subdivision depths of binary SSs for curves. The associated constants  $D_{k_0}, k_0 \geq 1$  defined in (15) of some binary subdivision curves are shown in Table 1.

**Table 1.** Associated constants of binary subdivision schemes (SSs).

Scheme/ $D_{k_0}$	$D_1 = \delta$	$D_2$	$D_3$	$D_4$	$D_5$
2-point scheme [14]	0.50000	0.25000	0.12500	0.06250	0.03125
3-point scheme [15]	1.50000	1.03125	0.83203	0.52685	0.36584
4-point scheme [16]	0.65625	0.36829	0.21610	0.12153	0.06912
4-point scheme [17]	0.80800	0.55800	0.40343	0.28765	0.20595
6-point scheme [18]	0.74200	0.44218	0.28589	0.18321	0.11768

**Example 1.** Consider the 2-point Chaikin’s binary SS [14]:

$$\begin{aligned}
 p_{2i}^{k+1} &= \frac{3}{4}p_i^k + \frac{1}{4}p_{i+1}^k, \\
 p_{2i+1}^{k+1} &= \frac{1}{4}p_i^k + \frac{3}{4}p_{i+1}^k.
 \end{aligned}
 \tag{28}$$

Its subdivision depth by Theorem 3 for  $D_{k_0}, k_0 \geq 1$  (computed in Table 1) are shown in Table 2. From this table, we see that as  $k_0$  increases the subdivision depth decreases. This shows that the sharp depth can be obtained by using our technique. In other words, we need a smaller number of iterations to get the sharp depth compared to the technique of [11]. For example, by [11], it needs thirty one iterations for  $\epsilon = 2.40 \times 10^{-11}$ , but using our technique, it needs only six iterations corresponding to  $D_5$ , which are shown in Figure 1a.

**Table 2.** Subdivision depth of 2-point binary subdivision curve.

$D_{k_0}/\epsilon$	0.0008	0.00002	$7.87 \times 10^{-7}$	$2.46 \times 10^{-8}$	$7.69 \times 10^{-10}$	$2.40 \times 10^{-11}$
$D_1 = \delta$	6	11	16	21	26	31
$D_2$	3	5	8	10	13	15
$D_3$	1	3	5	7	8	10
$D_4$	1	3	4	5	6	8
$D_5$	1	2	3	4	5	6

**Example 2.** Consider the 3-point approximating SS [15]:

$$\begin{aligned}
 f_{2i}^{k+1} &= \frac{1}{2}f_{i+1}^k + \frac{1}{2}f_{i+2}^k, \\
 f_{2i+1}^{k+1} &= \frac{1}{8}f_i^k + \frac{3}{4}f_{i+1}^k + \frac{1}{8}f_{i+2}^k.
 \end{aligned}
 \tag{29}$$

From Table 3,  $D_1 > 1$  and  $D_2 > 1$ , so error bounds cannot be computed by [4]. Consequently, it is not possible to compute subdivision depths. In this case, we increase the value of  $k_0$  until  $D_{k_0} < 1$ . The subdivision depths of the 3-point scheme are shown in Table 3 at different values of error tolerance. It is also demonstrated with the help of Figure 1b.

**Table 3.** Subdivision depth of 3-point binary subdivision curve.

$D_{k_0}/\epsilon$	0.0086	0.0031	0.0011	0.0004	0.0001	$5.67 \times 10^{-5}$	$2.07 \times 10^{-5}$
$D_3$	19	25	30	36	41	47	52
$D_4$	4	5	7	9	10	12	13
$D_5$	2	3	4	5	6	7	8

**Example 3.** The initial polygon have values  $f_i^k, k \geq 1$  by following the 4-point approximating binary SS [14] with subdivision mask

$$\begin{aligned} (a_{0,0}, a_{0,1}, a_{0,2}, a_{0,3}) &= \left( -\frac{7}{128}, \frac{105}{128}, \frac{35}{128}, -\frac{5}{128} \right), \\ (a_{1,0}, a_{1,1}, a_{1,2}, a_{1,3}) &= \left( -\frac{5}{128}, \frac{35}{128}, \frac{105}{128}, -\frac{7}{128} \right). \end{aligned} \tag{30}$$

Its subdivision depths by using Theorem 3 are given in Table 4. The comparison of the first and fifth convolution results is shown in Figure 1c.

**Table 4.** Subdivision depth of 4-point binary subdivision curve.

$D_{k_0}/\epsilon$	0.0092	0.0006	$4.43 \times 10^{-5}$	$3.06 \times 10^{-6}$	$2.11 \times 10^{-7}$	$1.46 \times 10^{-8}$
$D_1$	9	16	21	28	34	40
$D_2$	3	6	8	11	14	16
$D_3$	2	4	5	7	9	11
$D_4$	1	3	4	5	6	8
$D_5$	1	2	3	4	5	6

**Example 4.** Consider  $f_i^0 = f_i, i \in \mathbb{Z}$  denotes the initial polygon with values  $f_i^k, k \geq 1$ , which can be expressed recursively by the 4-point interpolating binary SS [17] with weights

$$\begin{aligned} (a_{0,0}, a_{0,1}, a_{0,2}, a_{0,3}) &= (0, 1, 0, 0), \\ (a_{1,0}, a_{1,1}, a_{1,2}, a_{1,3}) &= \left( -w, \frac{1}{2} + w, \frac{1}{2} + w, -w \right). \end{aligned} \tag{31}$$

Its subdivision depths are given in Table 5. It is also illustrated with the help of the graph given in Figure 1d.

**Table 5.** Subdivision depth of 4-point binary subdivision curve.

$D_{k_0}/\epsilon$	0.0259	0.0053	0.00109	0.0002	$4.66 \times 10^{-5}$	$9.59 \times 10^{-6}$	$1.97 \times 10^{-6}$
$D_1$	14	21	29	36	44	51	59
$D_2$	4	6	9	12	15	17	20
$D_3$	2	4	6	7	9	11	12
$D_4$	1	3	4	5	6	8	9
$D_5$	1	2	3	4	5	6	7

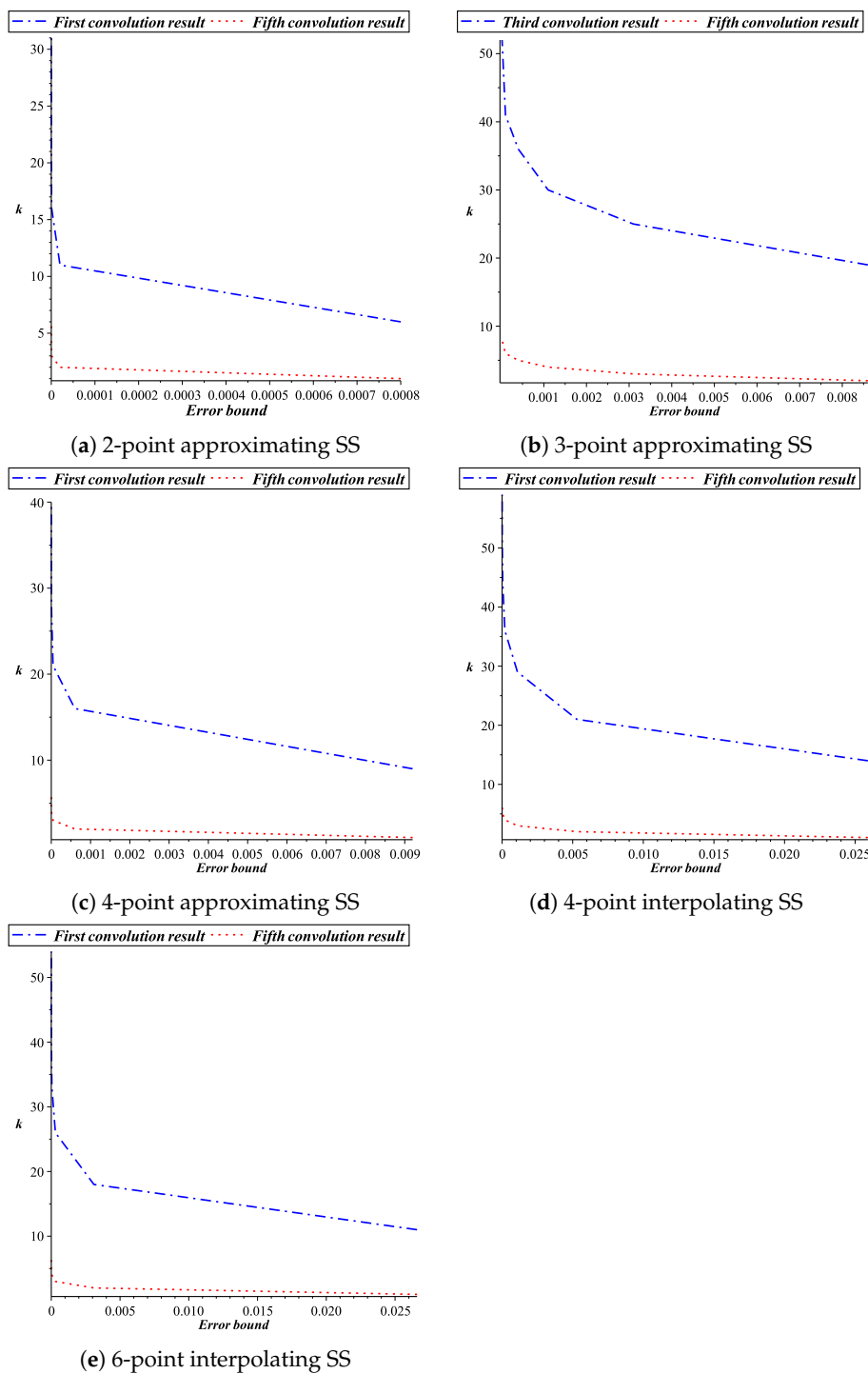
**Example 5.** Given the initial polygon  $f_i^0 = f_i, i \in \mathbb{Z}$  with values  $f_i^k, k \geq 1$  can be expressed recursively by the 6-point interpolating binary SS [18] as

$$\begin{aligned} (a_{0,0}, a_{0,1}, a_{0,2}, a_{0,3}, a_{0,4}, a_{0,5}) &= (0, 0, 1, 0, 0, 0), \\ (a_{1,0}, a_{1,1}, a_{1,2}, a_{1,3}, a_{1,4}, a_{1,5}) &= \left( w, -\frac{1}{16} - 3w, \frac{9}{16} + 2w, \frac{9}{16} + 2w, -\frac{1}{16} - 3w, w \right). \end{aligned} \tag{32}$$

In Table 6, subdivision depths are presented and its performance is shown in Figure 1e.

**Table 6.** Subdivision depth of 6-point binary subdivision curve.

$D_{k_0}/\epsilon$	0.0266	0.0031	0.0003	$4.35 \times 10^{-5}$	$5.12 \times 10^{-6}$	$6.02 \times 10^{-7}$	$7.09 \times 10^{-8}$
$D_1$	11	18	26	33	40	47	54
$D_2$	3	6	8	11	14	16	19
$D_3$	2	4	5	7	9	10	12
$D_4$	1	3	4	5	6	8	9
$D_5$	1	2	3	4	5	6	7



**Figure 1.** The performance of different subdivision schemes for univariate case. Here  $k$  presents the subdivision depth (level of subdivision process) obtained after user-specified error tolerance.

#### 4. Subdivision Depth for Binary Subdivision Surfaces

In the following section, firstly we calculate error bounds for subdivision surfaces. Secondly, we use these error bounds to compute subdivision depths.

**Theorem 4.** Consider  $f_{i,j}^0, i, j \in \mathbb{Z}$  to be the initial polygon and  $f_{i,j}^k$  to be the values for all non-negative integers recursively given in (5) along with the condition (6). Also let  $F^k$  be the representation of the polygon at the



points  $f_{i,j}^k$ . Then the error bounds of two successive refinements between the level  $k$  and  $k + 1$  by using the same technique given in [4] is

$$\|F^{k+1} - F^k\|_\infty \leq (\eta\beta_1 + \tau\beta_2 + \zeta\beta_3) F_{k_0} G_{k_0}, \tag{33}$$

where  $F_{k_0}, G_{k_0}, k_0 \geq 1$  defined in (21) and (22),  $\beta_t = \max_{i,j} \|\Delta_{i,j}^{0,t}\|, t = 1,2,3,$

$$\Delta_{i,j}^{k,1} = p_{i+1,j}^k - p_{i,j}^k, \quad \Delta_{i,j}^{k,2} = p_{i,j+1}^k - p_{i,j}^k, \quad \Delta_{i,j}^{k,3} = p_{i+1,j+1}^k - p_{i,j+1}^k,$$

and  $\eta, \tau$  and  $\zeta$  are defined by

$$\eta_1 = |a_{0,0}| \left( \sum_{r=1}^t |a_{0,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{0,q}| \right), \quad \eta_2 = |a_{0,0}| \left( \sum_{r=1}^t |a_{1,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| \right) + \frac{1}{2},$$

$$\eta_3 = |a_{1,0}| \left( \sum_{r=1}^t |a_{0,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{0,q}| \right), \quad \eta_4 = |a_{1,0}| \left( \sum_{r=1}^t |a_{1,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| \right) + \frac{1}{4},$$

$$\tau_1 = \sum_{r=1}^t |a_{0,r}| + \sum_{m=0}^t |a_{0,m}| \sum_{q=1}^{t-1} |\tilde{a}_{0,q}|, \quad \tau_2 = \sum_{r=1}^t |a_{0,r}| + \sum_{m=0}^t |a_{1,m}| \sum_{q=1}^{t-1} |\tilde{a}_{0,q}|,$$

$$\tau_3 = \sum_{r=1}^t |a_{1,r}| + \sum_{m=0}^t |a_{0,m}| \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| + \frac{1}{2}, \quad \tau_4 = \sum_{r=1}^t |a_{1,r}| + \sum_{m=0}^t |a_{1,m}| \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| + \frac{1}{2},$$

$$\zeta_1 = \sum_{r=1}^t |a_{0,r}| \left( \sum_{r=1}^t |a_{0,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{0,q}| \right), \quad \zeta_2 = \sum_{r=1}^t |a_{0,r}| \left( \sum_{r=1}^t |a_{1,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| \right),$$

$$\zeta_3 = \sum_{r=1}^t |a_{1,r}| \left( \sum_{r=1}^t |a_{0,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{0,q}| \right), \quad \zeta_4 = \sum_{r=1}^t |a_{1,r}| \left( \sum_{r=1}^t |a_{1,r}| + \sum_{q=1}^{t-1} |\tilde{a}_{1,q}| \right) + \frac{1}{4}.$$

Finally,

$$\eta = \max\{\eta_t; t = 1,2,3,4\}, \quad \tau = \max\{\tau_t; t = 1,2,3,4\}, \quad \zeta = \max\{\zeta_t; t = 1,2,3,4\}.$$

**Theorem 5.** Under the same circumstances used in Theorem 4. Let  $F^\infty$  be the limit surface associated with the subdivision process. Then

$$\|F^\infty - F^k\|_\infty \leq (\eta\beta_1 + \tau\beta_2 + \zeta\beta_3) \left( \frac{(F_{k_0} G_{k_0})^k}{1 - F_{k_0} G_{k_0}} \right), \tag{34}$$

where  $k_0 \geq 1$  is a natural number, such that  $F_{k_0} G_{k_0} < 1$ .

**Remark 2.** Again  $F_1 G_1$  is also equal to  $\delta$  which is defined in [4].

**Theorem 6.** Let  $k$  be the subdivision depth and let  $\nabla^k$  be the error bound between binary subdivision surface  $F^\infty$  and its  $k$ -level control polygon  $F^k$ . For arbitrary  $\epsilon > 0$ , if

$$k \geq \log_{(F_{k_0} G_{k_0})^{-1}} \left( \frac{\psi}{\epsilon(1 - F_{k_0} G_{k_0})} \right), \tag{35}$$

then  $\nabla^k \leq \epsilon$ .

**Proof.** Let  $\nabla^k$  denote the distance between limit curve  $F^\infty$  and initial curve (polygon)  $F^k$  at the  $k$ -th level. Then

$$\nabla^k = \|F^\infty - F^k\|_\infty \leq (\eta\beta_1 + \tau\beta_2 + \zeta\beta_3) \left( \frac{(F_{k_0}G_{k_0})^k}{1 - F_{k_0}G_{k_0}} \right). \tag{36}$$

To obtain given error tolerance  $\epsilon > 0$  the subdivision depth  $k$  satisfies the following inequality:

$$k \geq \log_{(F_{k_0}G_{k_0})^{-1}} \left( \frac{\psi}{\epsilon(1 - F_{k_0}G_{k_0})} \right), \tag{37}$$

then  $\nabla^k \leq \epsilon$ .

This completes the proof.  $\square$

*Application for Bivariate Case*

Here, we present some numerical examples to compute subdivision depths for subdivision surfaces. The associated constants  $F_{k_0}G_{k_0}, k_0 \geq 1$  for some binary subdivision surfaces by using (21) and (22) are shown in Table 7. We see that the values of  $F_{k_0}G_{k_0}$  decrease with the increase of  $k_0$ . This is the advantage of our approach.

**Table 7.** Associated constants of binary SSs.

Scheme/ $F_{k_0}D_{k_0}$	$F_1G_1 = \delta$	$F_2G_2$	$F_3G_3$	$F_4G_4$	$F_5G_5$
2-point scheme [14]	0.50000	0.25000	0.12500	0.06250	0.03125
3-point scheme [15]	1.50000	1.00000	0.81250	0.60937	0.47265
4-point scheme [16]	0.77930	0.40902	0.20020	0.09959	0.04953
4-point scheme [17]	0.84500	0.43741	0.22635	0.11557	0.05801
6-point scheme [18]	0.89780	0.45218	0.23205	0.11764	0.05902

**Example 6.** Given the initial polygon  $F_{i,j}^0 = F_{i,j}, i, j \in \mathbb{Z}$  and the values  $F_{i,j}^k, k \geq 1$  be described recursively by the tensor product of (28), then the subdivision depths for  $F_{k_0}G_{k_0}, k_0 \geq 1$  by using Theorem 35 are shown in Table 8 and the comparison of the first and fifth convolution result is shown in Figure 2a.

**Table 8.** Subdivision depth of 2-point binary subdivision surface.

$F_{k_0}G_{k_0}/\epsilon$	0.0020	0.00006	$1.96 \times 10^{-6}$	$6.15 \times 10^{-8}$	$1.92 \times 10^{-9}$	$6.008 \times 10^{-11}$
$F_1G_1$	6	11	16	21	26	31
$F_2G_2$	3	5	8	10	13	15
$F_3G_3$	2	3	5	7	8	10
$F_4G_4$	1	3	4	5	6	8
$F_5G_5$	1	2	3	4	5	6

**Example 7.** Consider the tensor product of the 3-point approximating SS defined in (29). The mask of SS satisfies  $F_{k_0}G_{k_0} < 1$ , for  $k_0 \geq 3$ . Therefore its subdivision depths can be computed at different values of error tolerance which are shown in Table 9. Here, the comparison of the third and fifth convolutions is demonstrated in Figure 2b.

**Table 9.** Subdivision depth of the 3-point binary subdivision surface.

$F_{k_0}G_{k_0}/\epsilon$	0.2688	0.1270	0.06007	0.0283	0.0134	0.0063	0.0029
$F_3G_3$	9	12	16	19	23	27	30
$F_4G_4$	2	4	5	7	8	10	11
$F_5G_5$	1	2	3	4	5	6	7

**Example 8.** Given that  $f_{i,j}^0 = f_{i,j}$ ,  $i, j \in \mathbb{Z}$ , let the values  $f_{i,j}^k$ ,  $k \geq 1$  be described recursively by the tensor product of (30). Then the subdivision depths for  $F_{k_0}G_{k_0}$ ,  $k_0 \geq 1$  by using Theorem 35 are shown in Table 10 which is illustrated in Figure 2c.

**Table 10.** Subdivision depth of 4-point binary subdivision surface.

$F_{k_0}G_{k_0}/\epsilon$	0.0192	0.0009	$4.72 \times 10^{-5}$	$2.33 \times 10^{-6}$	$1.15 \times 10^{-7}$	$5.73 \times 10^{-9}$
$F_1G_1$	18	30	42	54	66	78
$F_2G_2$	4	7	10	14	17	21
$F_3G_3$	2	4	6	8	9	11
$F_4G_4$	1	3	4	5	7	8
$F_5G_5$	1	2	3	4	5	6

**Example 9.** Consider the initial control polygon  $f_{i,j}^0 = f_{i,j}$ ,  $i, j \in \mathbb{Z}$  and let the values  $f_{i,j}^k$  for all positive integers be defined recursively by the tensor product of (31). Then the subdivision depths for  $F_{k_0}G_{k_0}$ ,  $k_0 \geq 1$  are shown in Table 11. It is also presented with the help of the graph in Figure 2d.

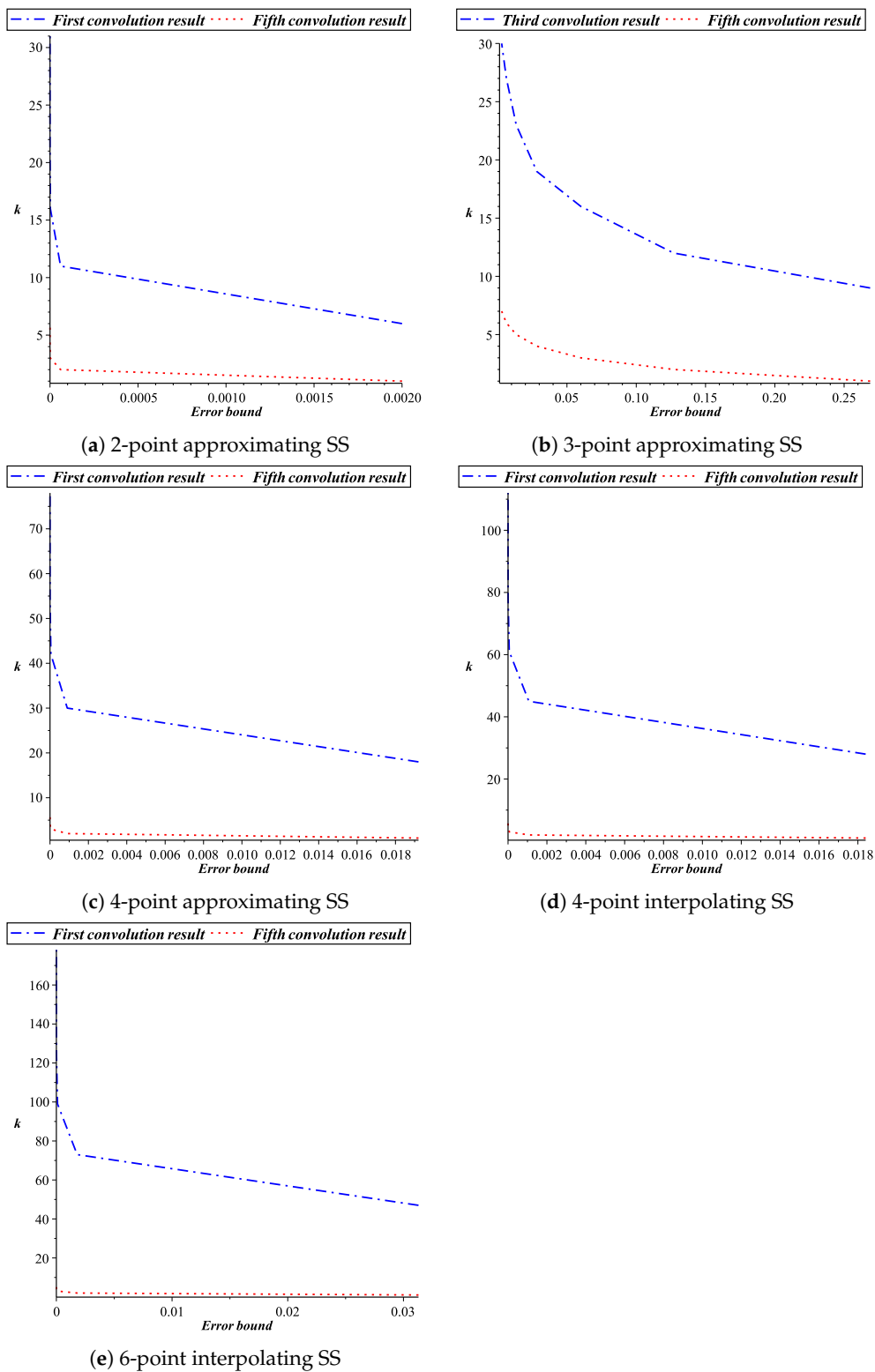
**Table 11.** Subdivision depth of 4-point binary subdivision surface.

$F_{k_0}G_{k_0}/\epsilon$	0.0184	0.00107	$6.21 \times 10^{-5}$	$3.6 \times 10^{-6}$	$2.09 \times 10^{-7}$	$1.21 \times 10^{-8}$
$F_1G_1$	28	45	61	78	95	112
$F_2G_2$	4	7	11	14	18	21
$F_3G_3$	2	4	6	8	10	12
$F_4G_4$	1	3	4	5	7	8
$F_5G_5$	1	2	3	4	5	6

**Example 10.** Given the initial control polygon  $f_{i,j}^0 = f_{i,j}$ ,  $i, j \in \mathbb{Z}$  and that the values  $f_{i,j}^k$ ,  $k \geq 1$  can be described recursively by the tensor product of (32), then in Table 12, subdivision depths for  $F_{k_0}G_{k_0}$ ,  $k_0 \geq 1$  are presented and a graphical representation is shown in Figure 2e.

**Table 12.** Subdivision depth of 6-point binary subdivision surface.

$F_{k_0}G_{k_0}/\epsilon$	0.0313	0.0018	0.0001	$6.44 \times 10^{-6}$	$3.80 \times 10^{-7}$	$2.24 \times 10^{-8}$
$F_1G_1$	47	73	99	126	152	178
$F_2G_2$	4	8	11	15	18	22
$F_3G_3$	2	4	6	8	10	12
$F_4G_4$	1	3	4	5	7	8
$F_5G_5$	1	2	3	4	5	6



**Figure 2.** Presents the performance of different subdivision schemes for bivariate case. Here  $k$  presents the subdivision depth (level of subdivision process) obtained after user-specified error tolerance.

### 5. Conclusions

In this paper, the modified version of the numerical algorithm of [4,11,12] have been presented to compute the subdivision depths of the SSs for curves and surfaces design. Our algorithm can be used

when the computation techniques of [4,11] fail. Moreover, we a smaller number of iterations to get the sharp subdivision depth compared to the existing algorithm. The proposed method is an efficient way of estimating subdivision depths for binary subdivision curves and surfaces.

**Author Contributions:** Conceptualization, F.K. and K.S.N.; Formal analysis, G.M. and D.B.; Methodology, A.G. and D.B.; Supervision, D.B.; Writing—original draft, A.S., F.K. and G.M.; Writing—review & editing, A.G. and K.S.N. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Barnhill, R.E.; Riesenfeld, R.F. *Computer Aided Geometric Design*; Academic Press: New York, NY, USA, 1974.
2. Faux, I.; Pratt, M. *Computational Geometry for Design and Manufacture*; Ellis Horwood: New York, NY, USA, 1979.
3. Huawei, W.; Youjiang, G.; Kaihuai, Q. Error estimation for Doo-Sabin surfaces. *Prog. Nat. Sci.* **2002**, *12*, 697–700.
4. Mustafa, G.; Chen, F.; Deng, J. Estimating error bounds for binary subdivision curves/surfaces. *J. Comput. Appl. Math.* **2006**, *193*, 596–613. [[CrossRef](#)]
5. Mustafa, G.; Hashmi, S.; Faheem, K. Estimating error bounds for non-stationary binary subdivision curves/surfaces. *J. Inf. Comput. Sci.* **2007**, *2*, 179–190.
6. Mustafa, G.; Deng, J. Estimating error bounds for ternary subdivision curve/surfaces. *J. Comput. Math.* **2007**, *25*, 473–484.
7. Mustafa, G.; Hashmi, S. Estimating error bounds for quaternary subdivision schemes. *J. Math. Anal. Appl.* **2009**, *358*, 159–167.
8. Mustafa, G.; Hashmi, S.; Noshi, N.A. Estimating error bounds for tensor product binary subdivision volumetric model. *Int. J. Comput. Math.* **2006**, *83*, 879–903. [[CrossRef](#)]
9. Mustafa, G.; Hashmi, S.; Faheem, K. Error bounds for  $\sqrt{3}$  subdivision surfaces. *J. Univ. Sci. Technol. China* **2009**, *39*, 1–9.
10. Mustafa, G.; Hashmi, S.; Faheem, K. Subdivision depth for triangular surfaces. *Alex. Eng. J.* **2016**, *55*, 1647–1653. [[CrossRef](#)]
11. Mustafa, G.; Hashmi, S. Subdivision depth computation for  $n$ -ary subdivision curves/surfaces. *J. Vis. Comput.* **2010**, *26*, 841–851. [[CrossRef](#)]
12. Moncayo, M.; Amat, S. Error bounds for a class of subdivision schemes based on the two-scale refinement equation. *J. Comput. Appl. Math.* **2011**, *236*, 265–278. [[CrossRef](#)]
13. Dyn, N.; Gregory, A.; Levin, D. Analysis of uniform binary subdivision scheme for curve design. *Constr. Approx.* **1991**, *7*, 127–147. [[CrossRef](#)]
14. Chaikin, G.M. An algorithm for high speed curve generation. *Comput. Graph. Image Process.* **1974**, *3*, 346–349. [[CrossRef](#)]
15. Dyn, N.; Levin, D.; Micchelli, C.A. Using parameters to increase smoothness of curves and surfaces generated by subdivision. *Comput. Aided Geom. Des.* **1990**, *7*, 129–140. [[CrossRef](#)]
16. Dyn, N.; Floater, M.S.; Hormann, K. A four-point subdivision scheme with fourth order accuracy and its extensions. In *Mathematical Methods for Curves and Surfaces: Modern Methods in Mathematics*; Springer: Troms, Norway, 2004 ; pp. 145–156.
17. Dyn, N.; Levin, D.; Gregory, J.A. A 4-Point Interpolatory SS for Curve Design. *Comput. Aided Geom. Des. Acad. Press.* **1987**, *4*, 257–268 [[CrossRef](#)]
18. Weissman, A. A 6-Point Interpolatory Subdivision Scheme for Curve Design. Master's Thesis, Tel Aviv University, Tel Aviv-Yafo, Israel, 1990.



Copyright of Symmetry (20738994) is the property of MDPI Publishing and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.