Available at
http://pvamu.edu/aam
Appl. Appl. Math.
ISSN: 1932-9466

Vol. 8, Issue 2 (December 2013), pp. 614 – 630

Applications and Applied Mathematics:
An International Journal
(AAM)

Numerical Algorithm for Analysis of *n*-ary Subdivision Schemes

Ghulam Mustafa and Muhammad Zahid
Department of Mathematics
The Islamia University of Bahawalpur
ghulam.mustafa@iub.edu; zahid_pak80@yahoo.com

Received: February 26, 2013; Accepted: August 21, 2013

Abstract

The analysis for continuity of limit curves generated by m-point n-ary subdivision schemes is presented for m, $n \geq 2$. The analysis is based on the study of corresponding differences and divided difference schemes. A numerical algorithm is introduced which computes the continuity and higher order divided differences of schemes in an efficient way. It is also free from polynomial factorization and division unlike the well-known Laurent polynomial algorithm for analysis of schemes which depends on polynomial algebraic operations. It only depends on the arithmetic operations.

Keywords: Subdivision scheme; divided difference; continuity; analysis; Laurent polynomial; numerical algorithm

AMS-MSC 2010 No.: 65D17, 65D07, 65D05.

1. Introduction

Computer aided geometric design is the branch of computational geometry which deals with the algorithms for designing smooth curves, surfaces and volumes. There is a very close relationship between computer aided geometric design and geometric modeling. The most common thing in computer aided design is the construction and representation of free form curves and surfaces by the set of points using polynomials.

Subdivision defines a curve or surface from an initial control mesh by recursive refinement. Thus subdivision schemes are widely used in computer graphics and computer aided geometric design for generating smooth curves and surfaces from discrete set of data points as they provide an efficient and flexible way for this purpose. The continuity of limit curve generated by a subdivision scheme is very important. So every scheme, when it is constructed, must be analyzed i.e. what is the order of continuity of the limit curve generated by this constructed scheme.

Dyn (2002) presented the technique for analysis of binary schemes by the formalism of Laurent polynomials. Later on this method was extended for ternary and quaternary schemes [Hassan and Dodgson (2001), Mustafa and Khan (2009)]. By algebraic operations on such a polynomial, sufficient conditions for convergence of the subdivision scheme, and for the smoothness of the limit curve generated by the subdivision scheme, can be checked rather automatically. Given the Laurent polynomial $a(z)$ of an n-ary subdivision scheme S_a, extended form of Laurent Polynomial Algorithm (LPA) for higher arity schemes can be restated as:

Step-1: If either $\sum_{j\in Z} a_{nj}, \sum_{j\in Z} a_{nj+1}, \ldots, \sum_{j\in Z} a_{nj+(n-1)} \neq 1$, the scheme does not converge. Stop!

Step-2: Compute $q(z) = \dfrac{a(z)}{1+z+z^2+\ldots+z^{n-1}}$.

Step-3: Set $q_1(z) = \sum_i q_i^{[1]} z^i = q(z)$.

Step-4: For $L = 1, \ldots, M$:

(a) Compute $N_L = \max_{0\le i\le n^L-1} \sum_j \left| q_{i-n^L j}^{[L]} \right|$.

(b) If $N_L < 1$, S_a is convergent. Stop!

(c) If $N_L \geq 1$, compute $q_{L+1}(z) = q(z)\, q_L(z^2) = \sum_i q_i^{[L+1]} z^i$.

Step-5: S_q is not contractive after M iterations. Stop!

Another but very old method for analysis of schemes is Divided Difference Algorithm (DDA) which was introduced for 4-point binary scheme by Dyn et al. (1987) and then its generalized version for m-point binary schemes by Dyn et al. (1991). Currently, DDA is not commonly used for analysis of the schemes. LPA involves polynomial factorization and division. Therefore in the analysis of m-point schemes with higher arity [Lian (2009), Mustafa and Rehman (2010), Zheng et al. (2009)] by LPA the need to handle higher order polynomials and their factorization and division, has motivated us to introduce an algorithm for analysis which should be free from polynomial algebraic operations.

According to Dyn et al. (1991) “If the n^{th} order divided difference of the original binary scheme is C^0-continuous then the original binary scheme will be C^n-continuous”. Moreover, according to Sabin (2010), “Higher order divided differences are just divided differences of divided differences". In this article, we have generalized these ideas into numerical algorithms for divided differences and the analysis of the *m*-point *n*-ary subdivision schemes.

Contributions: The main contributions of the paper are

- Numerical algorithm for divided differences of *m*-point *n*-ary schemes.
- Numerical algorithm for continuity of *m*-point *n*-ary subdivision schemes. In this algorithm, we have replaced Step-2 and Step-3 of LPA by simple arithmetic operations.
- A demonstration by numerical examples that Proposed Numerical Algorithm (PNA) and LPA give the same results.

The rest of the paper is organized as follows: In Section 2, we discuss *n*-ary schemes, their divided differences and convergence. We also present numerical algorithm for divided differences in this section. Section 3 is dedicated to the smoothness analysis of the schemes. We present numerical algorithm for continuity of the schemes, numerical examples and comparison of PNA and LPA in this section.

2. *n*-ary and its Divided Difference Schemes

In this section, we present an *n*-ary scheme and compute the maximal difference between its two consecutive control polygons at different subdivision levels. Convergence of the *n*-ary scheme is also proved in this section. Its first order divided difference scheme is presented at the end of this section.

2.1. *n*-ary Univariate Schemes

Let $p_i^k, i \in Z,$ denote a sequence of points in $R^N, N \geq 2,$ where k is a non-negative integer. An *n*-ary subdivision process defined by Aspert (2003) is

$$p_{ni+q}^{k+1} = \sum_{j=0}^{m-1} a_{j,q}\, p_{i+j}^{k}, \qquad q = 0,1,2,\ldots,n-1., \tag{2.1}$$

where m, $n \geq 2$ and coefficients $\{a_{j,q}\}$ are called subdivision mask. If $a(z)$ is the Laurent polynomial of the above scheme then entries in $a(z)$ and coefficients $\{a_{j,q}\}$ are related as $\{a_{j,q}\} = \{a_{nj+q}\}$. The necessary condition for uniform convergence of scheme (2.1) is

$$\sum_{j=0}^{m-1} a_{j,q} = 1, \qquad q = 0,1,2,\ldots,n-1. \tag{2.2}$$

2.2. Maximal Differences between Control Polygons

In this section, we compute the maximal difference between the $(k+1)^{st}$ level control polygon $p^{k+1}=\{p_i^{k+1}\}$ and k^{th} level control polygon p^k n-ary subdivision scheme (2.1).

Lemma 2.1.

Given an initial control polygon $p_i^0 = p_i, i \in Z,$ let the values $p_i^k, k \geq 1$ be defined recursively by subdivision process (2.1) together with (2.2) then

$$\max_i \left\| p_{i+1}^{k+1} - p_i^{k+1} \right\| \leq (\beta)^{k+1} \max_i \left\| p_{i+1}^0 - p_i^0 \right\|, \tag{2.3}$$

where

$$\beta = \max\{\beta_r , \beta_{n-1}\}, \quad q = 0,1,2,...,n-2, \tag{2.4}$$

$$\beta_r = \sum_{l=0}^{m-2} \left| \sum_{j=0}^{l} (a_{j,r} - a_{j,r+1}) \right|, \quad r = 0,1,2,...,n-2, \tag{2.5}$$

$$\beta_{n-1} = \sum_{l=0}^{m-1} \left| \sum_{j=0}^{l} (a_{j,n-1} - a_{j-1,0}) \right|,$$

where n is the arity of the scheme and $a_{-1,0}$ is zero throughout the paper.

Proof:

Using (2.1), we get

$$p_{ni+(r+1)}^{k+1} - p_{ni+r}^{k+1} = \sum_{j=0}^{m-1} \left\{ (a_{j,r+1} - a_{j,r})\, p_{i+j}^k \right\},$$

that implies

$$p_{ni+(r+1)}^{k+1} - p_{ni+r}^{k+1} = (a_{0,r} - a_{0,r+1})(p_{i+1}^k - p_i^k) + (a_{0,r} + a_{1,r} - a_{0,r+1} - a_{1,r+1})(p_{i+2}^k - p_{i+1}^k)$$
$$+A+B,$$

where

$$A = (a_{0,r} + a_{1,r} + a_{2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1})(p_{i+3}^k - p_{i+2}^k) + ... + (a_{0,r} + a_{1,r} + a_{2,r}$$
$$+ ... + a_{m-2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1} - ... - a_{m-2,r+1})(p_{i+m-1}^k - p_{i+m-2}^k),$$

and

$$B=(-a_{0,r}-a_{1,r}-a_{2,r}-\ldots-a_{m-1,r}+a_{0,r+1}+a_{1,r+1}+a_{2,r+1}+\ldots+a_{m-1,r+1})p^k_{i+m-1}.$$

Further it implies that

$$p^{k+1}_{ni+(r+1)}-p^{k+1}_{ni+r}=\sum_{l=0}^{m-2}\left\{\sum_{j=0}^{l}(a_{j,r}-a_{j,r+1})(p^k_{i+l+1}-p^k_{i+l})\right\}+\sum_{j=0}^{m-1}(a_{j,r+1}-a_{j,r})p^k_{i+m-1}\,.$$

Since by (2.2), $\sum_{j=0}^{m-1}(a_{j,r+1}-a_{j,r})=0$ for $r=0,1,2,\ldots,n-2$, so

$$p^{k+1}_{ni+(r+1)}-p^{k+1}_{ni+r}=\sum_{l=0}^{m-2}\left\{\sum_{j=0}^{l}(a_{j,r}-a_{j,r+1})(p^k_{i+l+1}-p^k_{i+l})\right\}.$$

Let $\|\cdot\|_\infty$ denote the maximum norm, then

$$\left\|p^{k+1}_{ni+(r+1)}-p^{k+1}_{ni+r}\right\|_\infty\le\beta_r\max_i\left\|p^k_{i+1}-p^k_i\right\|, \tag{2.6}$$

where

$$\beta_r=\sum_{l=0}^{m-2}\left|\sum_{j=0}^{l}(a_{j,r}-a_{j,r+1})\right|,\qquad r=0,1,2,\ldots,n-2.$$

Similarly,

$$p^{k+1}_{ni+n}-p^{k+1}_{ni+(n-1)}=\sum_{l=0}^{m-1}\left\{\sum_{j=0}^{l}(a_{j,n-1}-a_{j-1,0})(p^k_{i+l+1}-p^k_{i+l})\right\}.$$

This implies

$$\left\|p^{k+1}_{ni+n}-p^{k+1}_{ni+(n-1)}\right\|_\infty\le\beta_{n-1}\max_i\left\|p^k_{i+1}-p^k_i\right\|, \tag{2.7}$$

where

$$\beta_{n-1}=\sum_{l=0}^{m-1}\left|\sum_{j=0}^{l}(a_{j,n-1}-a_{j-1,0})\right|.$$

Now by (2.6) and (2.7), we get (2.3). This completes the proof. □

By extending the technique given in Lemma 3.1 of Dyn et al. (1991), we get the following lemma.

Lemma 2.2.

Given an initial control polygon $p_i^0 = p_i, i \in Z,$ let the values $p_i^k, k \geq 1$ be defined recursively by subdivision process (2.1). Suppose p^k is the piecewise linear interpolant to the values p_i^k. Then, the maximal difference between p^{k+1} and p^k is

$$\left\| p^{k+1} - p^k \right\|_\infty \leq \beta^* \max_i \left\| p_{i+1}^k - p_i^k \right\|, \tag{2.8}$$

where β^* is some real number.

Theorem 2.1.

Given $p_i^0 = p_i, i \in Z,$ let the values $p_i^k, k \geq 1$ be defined recursively by subdivision process (2.1) together with the necessary condition (2.2) and p^k be the piecewise linear interpolant on $[0, n_1]$ to the values p_i^k. Then, for $\beta < 1$, where β is defined by (2.4), there exists $\lim_{k \to \infty} p^k = p \in C[0, n_1]$ which means scheme (2.1) is C^0-continuous.

Proof:

Consider the piecewise linear interpolant p^k on $[0, n_1]$ to the values p_i^k and let $\|\cdot\|_\infty$ denote the uniform norm on $C[0, n_1]$. We will show that $\{p^k\}_{k=0}^\infty$ defines a Cauchy sequence on $C[0, n_1]$. Since the maximal difference between p^{k+1} and p^k is attained at a point on the $(k+1)^{st}$ mesh, then by (2.8), we get

$$\left\| p^{k+1} - p^k \right\|_\infty \leq \beta^* \max_i \left\| p_{i+1}^k - p_i^k \right\|,$$

where β^* is some real number. Utilizing (2.3), we get

$$\left\| p^{k+1} - p^k \right\|_\infty \leq \beta^* \left(\beta^k\right) \max_i \left\| p_{i+1}^0 - p_i^0 \right\|,$$

where β is defined by (2.4). If $\beta < 1$, then it follows that $\{p^k\}_{k=0}^\infty$ defines a Cauchy sequence on $C[0, n_1]$ and

$$\lim_{k \to \infty} p^k = p \in C[0, n_1].$$

This completes the proof. □

***Remark* 2.1.**

Here we note that, $\beta < 1$ is a sufficient condition for C^0-continuity of the n-ary scheme where β is defined by (2.4) and (2.5).

2.3. First order divided difference scheme

Lemma 2.3.

The first order divided difference process of scheme defined by (2.1) is defined as

$$d_{ni+r}^{k+1} = n\sum_{l=0}^{m-2}\left\{\sum_{j=0}^{l}(a_{j,r} - a_{j,r+1})\, d_{i+l}^{k}\right\}, \qquad r = 0,1,2,\ldots,n-2,$$

$$d_{ni+(n-1)}^{k+1} = \sum_{l=0}^{m-1}\left\{\sum_{j=0}^{l}(a_{j,n-1} - a_{j-1,0})\, d_{i+l}^{k}\right\}. \tag{2.9}$$

Proof:

If p^k is the piecewise linear interpolant to the values p_j^k then the first order divided difference is given by

$$d_j^k = n^k \Delta p_j^k = n^k\left(p_{j+1}^k - p_j^k\right). \tag{2.10}$$

By replacing j by $ni+r$ and k by $k+1$ in above equation, we get

$$d_{ni+r}^{k+1} = n^{k+1}(p_{ni+(r+1)}^{k+1} - p_{ni+r}^{k+1}).$$

Using (2.1), we get

$$\begin{aligned} d_{ni+r}^{k+1} = n^{k+1}(&a_{0,r+1}p_i^k + a_{1,r+1}p_{i+1}^k + a_{2,r+1}p_{i+2}^k + \ldots + a_{m-1,r+1}p_{i+m-1}^k \\ &- a_{0,r}p_i^k - a_{1,r}p_{i+1}^k - a_{2,r}p_{i+2}^k - \ldots - a_{m-1,r}p_{i+m-1}^k). \end{aligned}$$

Simplifying, we get

$$\begin{aligned} d_{ni+r}^{k+1} = n^{k+1}\{&(a_{0,r+1} - a_{0,r})p_i^k + (a_{1,r+1} - a_{1,r})p_{i+1}^k + (a_{2,r+1} - a_{2,r})p_{i+2}^k \\ &+ \ldots + (a_{m-1,r+1} - a_{m-1,r})p_{i+m-1}^k\}. \end{aligned} \tag{2.11}$$

We want to make an equation of the form

$$d_{ni+r}^{k+1} = x_0 d_i^k + x_1 d_{i+1}^k + x_2 d_{i+2}^k + \ldots + x_{m-2} d_{i+m-2}^k . \tag{2.12}$$

This implies

$$d_{ni+r}^{k+1} = x_0 n^k (-p_i^k + p_{i+1}^k) + x_1 n^k (-p_{i+1}^k + p_{i+2}^k) + x_2 n^k (-p_{i+2}^k + p_{i+3}^k) + \ldots + x_{m-2} n^k (-p_{i+m-2}^k + p_{i+m-1}^k) .$$

Simplifying, we get

$$d_{ni+r}^{k+1} = n^k \{-x_0 p_i^k + (x_0 - x_1) p_{i+1}^k + (x_1 - x_2) p_{i+2}^k + (x_2 - x_3) p_{i+3}^k + \ldots + (x_{m-3} - x_{m-2}) p_{i+m-2}^k + x_{m-2} p_{i+m-1}^k \} . \tag{2.13}$$

Comparing the coefficients of p_i^k 's in (2.11) and (2.13) and solving simultaneously, we get

$$\begin{aligned}
x_0 &= n\left(a_{0,r} - a_{0,r+1}\right),\\
x_1 &= n\left(a_{0,r} + a_{1,r} - a_{0,r+1} - a_{1,r+1}\right),\\
x_2 &= n\left(a_{0,r} + a_{1,r} + a_{2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1}\right),\\
&\vdots\\
x_{m-2} &= n\left(a_{0,r} + a_{1,r} + a_{2,r} + \ldots + a_{m-2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1} - \ldots - a_{m-2,r+1}\right).
\end{aligned}$$

Substituting in (2.12), we get

$$d_{ni+r}^{k+1} = n\{(a_{0,r} - a_{0,r+1}) d_i^k + (a_{0,r} + a_{1,r} - a_{0,r+1} - a_{1,r+1}) d_{i+1}^k + (a_{0,r} + a_{1,r} + a_{2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1}) d_{i+2}^k + \ldots + (a_{0,r} + a_{1,r} + a_{2,r} + \ldots + a_{m-2,r} - a_{0,r+1} - a_{1,r+1} - a_{2,r+1} - \ldots - a_{m-2,r+1}) d_{i+m-2}^k \} .$$

This implies

$$d_{ni+r}^{k+1} = n \sum_{l=0}^{m-2} \left\{ \sum_{j=0}^{l} (a_{j,r} - a_{j,r+1}) d_{i+l}^k \right\}, \qquad r = 0, 1, 2, \ldots, n-2 .$$

By replacing j by $ni + (n-1)$ and k by $k+1$ in (2.10), we get

$$d_{ni+(n-1)}^{k+1} = n^{k+1} (p_{ni+n}^{k+1} - p_{ni+(n-1)}^{k+1}) .$$

Using (2.1) and similar procedure as above, we get

$$d_{ni+(n-1)}^{k+1} = \sum_{l=0}^{m-1} \left\{ \sum_{j=0}^{l} (a_{j,n-1} - a_{j-1,0})\, d_{i+l}^{k} \right\}.$$

This completes the proof. □

Note:

Without loss of generality, we can vary $l = 0 \ldots m-1$ instead of $l = 0 \ldots m-2$ in the equations representing $\beta_r : (2.5)$ and $d_{ni+r}^{k+1} : (2.9)$ as the entry for $j = m-1$ is zero. This is just to avoid extra computation in algorithm.

2.4. Numerical Algorithm for Divided Differences

Since higher order divided differences are just divided differences of divided differences by Sabin (2010) utilizing (2.9) recursively, we get a numerical algorithm for computing higher order divided differences of *m*-point *n*-ary subdivision schemes for *m*, *n* ≥ 2. This algorithm is fast and efficient because simple algebraic operations in Steps 1-2 are performed on the right hand sides of equations and results assigned to the left hand side of the equations without using extra computer memory to save newly computed values.

Input: Enter the mask of the scheme $a_{j,q}$, $j = 0$ to $m-1$, $q = 0$ to $n-1$, where *m* and *n* stand for the complexity (i.e., number of points involved to insert a new point in the control polygon) and arity of the scheme respectively.

Step 1: Compute (*Fragment of* (2.5) *and* (2.9))

$$\begin{aligned} a_{j,n} &= a_{j,0}, && j = 0 \ldots m-1, \\ a_{j,r} &= a_{j-1,r} + a_{j,r} - a_{j,r+1}, && r = 0 \ldots n-2,\ j = 0 \ldots m-1, \\ a_{j,n-1} &= a_{j-1,n-1} + a_{j,n-1} - a_{j-1,n}, && j = 0 \ldots m-1, \end{aligned}$$

where $a_{-1,q} = 0$ for $q = 0 \ldots n$.

Step 2: Compute (*Fragment of* (2.9))

$$\begin{aligned} a_{j,r} &= n\, a_{j,r}, && r = 0 \ldots n-2,\ j = 0 \ldots m-1, \\ a_{j,n-1} &= n\, a_{j,n-1}, && j = 0 \ldots m-1. \end{aligned}$$

Goto Step-1, for next higher order divided difference otherwise exit.

Output: s^{th} order divided difference can be obtained by cycling *s*-times Steps 1-2.

3. Numerical Algorithm for Continuity of Scheme

Here we first summarize the above results and then present the numerical algorithm for continuity of *m*-point *n*-ary scheme. The necessary and sufficient conditions for C^0-continuity are given in (2.2) and Theorem 2.3 (i.e., $\beta < 1$) respectively. For higher order continuities we need higher order divided differences which can be computed by using the above proposed numerical algorithm. By Dyn et al. (1991) a given scheme will be C^0-continuous if its *n*th divided difference scheme is C^0-continuous. These results lead to establish following numerical algorithm for computing the order of continuity of the *m*-point *n*-ary subdivision scheme:

Input: Enter the mask of the scheme $a_{j,q}$, $j = 0$ to $m-1$, $q = 0$ to $n-1$, where *m* and *n* stand for the complexity and arity of the scheme respectively.

Step 1: *Utilization of* (2.2): If $\sum_{j=0}^{m-1} a_{j,q} = 1$, for $q = 0$ to $n-1$, then goto Step 2, otherwise exit.

Step 2: *Fragment of* (2.5) *and* (2.9), *which is common in both equations*:
Do computations on the right hand sides of the following equations and assign the results to the left hand sides of equations:

$$a_{j,n} = a_{j,0}, \qquad j = 0 \ldots m-1,$$
$$a_{j,r} = a_{j-1,r} + a_{j,r} - a_{j,r+1}, \qquad r = 0 \ldots n-2,\ j = 0 \ldots m-1,$$
$$a_{j,n-1} = a_{j-1,n-1} + a_{j,n-1} - a_{j-1,n}, \qquad j = 0 \ldots m-1,$$

where $a_{-1,q} = 0$ for $q = 0 \ldots n$.

Step 3: *Fragment of* (2.5) *and sufficient condition*: By using updated values $a_{j,r}$ and $a_{j,n-1}$ by Step 2, compute,

$$\beta_r = \sum_{j=0}^{m-1} \left| a_{j,r} \right| \quad \text{for} \quad r = 0 \ldots n-2, \text{ and } \beta_{n-1} = \sum_{j=0}^{m-1} \left| a_{j,n-1} \right|.$$

If $\beta = \max\{\beta_r, \beta_{n-1}\} < 1$ then goto Step 4, otherwise exit.

Step 4: *Fragment of* (2.9): By using updated values $a_{j,r}$ and $a_{j,n-1}$ by Step 2 in right hand sides of the following equations and assigning the results to the left hand sides of the equations, compute,

$$a_{j,r} = n\,a_{j,r}, \qquad r = 0 \ldots n-2,\ j = 0 \ldots m-1,$$
$$a_{j,n-1} = n\,a_{j,n-1}, \qquad j = 0 \ldots m-1.$$

Go to Step 1.

Output: The *s*-times successful completion of Steps 1-4 mean original scheme is C^s-continuous.

The validity of above algorithm has been checked by computing the continuity of some well-known schemes. The following results obtained by the proposed numerical algorithm coincide with the results obtained by the generalized Laurent polynomial algorithm.

Corollary 3.1.

Given $p_i^0 = p_i, i \in Z$, let the values p_i^k, $k \geq 1$ be defined recursively by following 4-point interpolating binary subdivision scheme introduced by Dyn et al. (1987)

$$\begin{aligned} p_{2i}^{k+1} &= p_i^k , \\ p_{2i+1}^{k+1} &= -\omega\, p_{i-1}^k + \left(\tfrac{1}{2}+\omega\right) p_i^k + \left(\tfrac{1}{2}+\omega\right) p_{i+1}^k - \omega\, p_{i+2}^k , \end{aligned} \tag{3.1}$$

then scheme (3.1) is C^1-continuous over the parametric interval $0 < \omega < 1/8$.

Proof:

Here $m = 4,\ n = 2$ and the mask of the scheme is:

$$\begin{array}{llll} a_{0,0} = 0, & a_{1,0} = 1, & a_{2,0} = 0, & a_{3,0} = 0, \\ a_{0,1} = -\omega, & a_{1,1} = \tfrac{1}{2}+\omega, & a_{2,1} = \tfrac{1}{2}+\omega, & a_{3,1} = -\omega . \end{array}$$

1st Round:

Step 1: Clearly $\sum_{j=0}^{3} a_{j,q} = 1,\ q = 0,1$.

Step 2:

$$\begin{aligned} & a_{0,2} = a_{0,0} = 0,\ a_{1,2} = a_{1,0} = 1, a_{2,2} = a_{2,0} = 0, a_{3,2} = a_{3,0} = 0, \\ & a_{0,0} = a_{0,0} - a_{0,1} = \omega, \\ & a_{1,0} = a_{0,0} + a_{1,0} - a_{1,1} = \tfrac{1}{2}, \\ & a_{2,0} = a_{1,0} + a_{2,0} - a_{2,1} = -\omega, \\ & a_{3,0} = a_{2,0} + a_{3,0} - a_{3,1} = 0, \\ & a_{0,1} = a_{0,1} = -\omega, \\ & a_{1,1} = a_{0,1} + a_{1,1} - a_{0,2} = \tfrac{1}{2}, \\ & a_{2,1} = a_{1,1} + a_{2,1} - a_{1,2} = \omega, \\ & a_{3,1} = a_{2,1} + a_{3,1} - a_{2,2} = 0 . \end{aligned}$$

Step 3:

$$\beta_0 = \sum_{j=0}^{3} \left| a_{j,0} \right| = \left| \omega \right| + \left| 1/2 \right| + \left| -\omega \right| + \left| 0 \right| ,$$

$$\beta_1 = \sum_{j=0}^{3} \left| a_{j,1} \right| = \left| -\omega \right| + \left| 1/2 \right| + \left| \omega \right| + \left| 0 \right| .$$

Since $\beta = \max\{\beta_0 , \beta_1\} < 1$, for $\omega \in \left(\frac{-3}{2}, \frac{1}{4}\right)$, therefore scheme (3.1) is C^0.

Step 4:

$$a_{0,0} = 2\omega, \quad a_{1,0} = 1, \quad a_{2,0} = -2\omega, \quad a_{3,0} = 0,$$
$$a_{0,1} = -2\omega, \quad a_{1,1} = 1, \quad a_{2,1} = 2\omega, \quad a_{3,1} = 0,$$

2nd Round:

Step 1: Clearly $\sum_{j=0}^{3} a_{j,q} = 1,\ q = 0,1$.

Step 2:

$$a_{0,2} = a_{0,0} = 2\omega,\ a_{1,2} = a_{1,0} = 1, a_{2,2} = a_{2,0} = -2\omega, a_{3,2} = a_{3,0} = 0,$$
$$a_{0,0} = a_{0,0} - a_{0,1} = 4\omega,$$
$$a_{1,0} = a_{0,0} + a_{1,0} - a_{1,1} = 4\omega,$$
$$a_{2,0} = a_{1,0} + a_{2,0} - a_{2,1} = 0,$$
$$a_{3,0} = a_{2,0} + a_{3,0} - a_{3,1} = 0,$$
$$a_{0,1} = a_{0,1} = -2\omega,$$
$$a_{1,1} = a_{0,1} + a_{1,1} - a_{0,2} = 1 - 4\omega,$$
$$a_{2,1} = a_{1,1} + a_{2,1} - a_{1,2} = -2\omega,$$
$$a_{3,1} = a_{2,1} + a_{3,1} - a_{2,2} = 0 .$$

Step 3:

$$\beta_0 = \sum_{j=0}^{3} \left| a_{j,0} \right| = \left| 4\omega \right| + \left| 4\omega \right| + \left| 0 \right| + \left| 0 \right| ,$$
$$\beta_1 = \sum_{j=0}^{3} \left| a_{j,1} \right| = \left| -2\omega \right| + \left| 1 - 4\omega \right| + \left| -2\omega \right| + \left| 0 \right| .$$

Since $\beta = \max\{\beta_0 , \beta_1\} < 1$, for $\omega \in \left(0, \frac{1}{8}\right)$, therefore first order divided difference scheme is C^0 while scheme (3.1) is C^1-continuous.

Step 4:

$$a_{0,0} = 8\omega, \quad a_{1,0} = 8\omega, \quad a_{2,0} = 0, \quad a_{3,0} = 0,$$
$$a_{0,1} = -4\omega, \quad a_{1,1} = 2 - 8\omega, \quad a_{2,1} = -4\omega, \quad a_{3,1} = 0.$$

3^{rd} Round:

Step-1: Clearly $\sum_{j=0}^{3} a_{j,q} \neq 1,\ q = 0,1,$ exit.

The scheme is C^1-continuous for $\omega \in \left(0, \frac{1}{8}\right)$.

Corollary 3.2.

The scheme (3.1) is C^1-continuous by alternating approach LPA.

Proof:

The Laurent polynomial $a(z)$ for the mask of the 4-point binary scheme can be written as

$$a(z) = -\omega z^{-3} + \left(\tfrac{1}{2} + \omega\right) z^{-1} + z^0 + \left(\tfrac{1}{2} + \omega\right) z^1 - \omega z^3 .$$

Now,

$$q(z) = \frac{a(z)}{1+z} = -\omega z^{-3} + \omega z^{-2} + \tfrac{1}{2} z^{-1} + \tfrac{1}{2} z^0 + \omega z^1 - \omega z^2 ,$$

and

$$\left\| S_q \right\|_\infty = \tfrac{1}{2} + 2 |\omega| .$$

As we know S_a is convergent (C^0 *-continuity*) iff $\left\| S_q \right\|_\infty < 1$ for some $L \in Z_+/0$. So for $L = 1$, $\left\| S_q \right\|_\infty < 1$, if $|\omega| < \frac{1}{4}$. Now by computing $q_2(z) = q(z)\ q(z^2)$, we get

$$\begin{aligned} q_2(z) = {} & \omega^2 z^{-9} - \omega^2 z^{-8} - (\tfrac{1}{2}\omega + \omega^2) z^{-7} + (\omega^2 - \tfrac{1}{2}\omega) z^{-6} - \omega^2 z^{-5} + (\omega + \omega^2) z^{-4} \\ & + (\tfrac{1}{4} + \omega^2 - \tfrac{1}{2}\omega) z + (\tfrac{1}{4} - \omega^2 + \tfrac{1}{2}\omega) z^{-2} + (\tfrac{1}{4} - \omega^2 + \tfrac{1}{2}\omega) z^{-1} + (\tfrac{1}{4} + \omega^2 - \tfrac{1}{2}\omega) \\ & + (\omega + \omega^2) z^1 - \omega^2 z^2 + (\omega^2 - \tfrac{1}{2}\omega) z^3 - (\tfrac{1}{2}\omega + \omega^2) z^4 - \omega^2 z^5 + \omega^2 z^5 . \end{aligned}$$

This leads to

$$\left\| S_q^2 \right\|_\infty = \max\left\{ \left|\tfrac{1}{2} + \omega\right| |\omega| + \left|\tfrac{1}{4} + \omega^2 - \tfrac{1}{2}\omega\right| + |1 + \omega| |\omega| + \omega^2 ,\right.$$

and

$$\left. \left|\omega - \tfrac{1}{2}\right| |\omega| + \left|\tfrac{1}{4} - \omega^2 + \tfrac{1}{2}\omega\right| + 2\omega^2 \right\}.$$

Thus, for the range $\frac{-3}{8} < \omega < \frac{-1+\sqrt{13}}{8} < \frac{1}{2}$ and $L = 2$, we have $\left\|S_q\right\|_\infty < 1$. As for smoothness analysis (C^1 -continuity), consider S_b with $b(z) = \frac{2a(z)}{1+z}$. Then, S_b is convergent iff S_r is contractive. Now,

$$r(z) = \frac{b(z)}{1+z} = \frac{2a(z)}{1+z} = 2z^{-3}\left(\frac{1}{2}z^2 - \omega(z-1)^2(1+z^2)\right).$$

But for the case $L = 1$, $\left\|S_r\right\|_\infty \geq 1$. Therefore, to see the contractivity, we consider $\left\|S_r\right\|_\infty$. The condition $\left\|S_r\right\|_\infty < 1$ gives the range $0 < \omega < \frac{-1+\sqrt{5}}{8} \cong 0.154$.

***Remark* 3.1.**

From Corollary 3.1 and 3.2, one can see that PNA is free from polynomial operations and depend on only arithmetic operations while LPA depends on polynomial as well as on arithmetic operations. So it is obvious that the computational complexity of PNA is less than the complexity of LPA.

Corollary 3.3.

Given $p_i^0 = p_i, i \in Z$, let the values p_i^k, $k \geq 1$ be defined recursively by the following 6-point interpolating ternary subdivision scheme introduced by Khan and Mustafa (2008)

$$p_{3i}^{k+1} = p_i^k,$$

$$\begin{aligned} p_{3i+1}^{k+1} &= (-\tfrac{11}{81}+13\omega)p_{i-2}^k + (\tfrac{13}{27}-51\omega)p_{i-1}^k + (-\tfrac{2}{27}+74\omega)p_i^k \\ &\quad + (\tfrac{74}{81}-46\omega)p_{i+1}^k + (-\tfrac{5}{27}+9\omega)p_{i+2}^k + \omega\, p_{i+3}^k, \end{aligned} \tag{3.2}$$

$$\begin{aligned} p_{3i+2}^{k+1} &= \omega\, p_{i-2}^k + (-\tfrac{5}{27}+9\omega)p_{i-1}^k + (\tfrac{74}{81}-46\omega)p_i^k \\ &\quad + (-\tfrac{2}{27}+74\omega)p_{i+1}^k + (\tfrac{13}{27}-51\omega)p_{i+2}^k + (-\tfrac{11}{81}+13\omega)\, p_{i+3}^k, \end{aligned}$$

then scheme (3.2) is C^2-continuous over the interval $\omega \in (\frac{14}{1215}, \frac{23}{1944})$.

Corollary 3.4.

Given $p_i^0 = p_i, i \in Z$, let the values p_i^k, $k \geq 1$ be defined recursively by the following 4-point approximating quaternary subdivision scheme introduced by Hassan and Dodgson (2001)

$$f_{4i}^{k+1} = (\tfrac{7}{32}-\tfrac{7}{64}\omega)f_{i-1}^k + (\tfrac{29}{64}+\tfrac{13}{64}\omega)f_i^k + (\tfrac{5}{16}-\tfrac{5}{64}\omega)f_{i+1}^k + (\tfrac{1}{64}-\tfrac{1}{64}\omega)f_{i+2}^k,$$

$$f_{4i+1}^{k+1} = (\tfrac{15}{128}-\tfrac{5}{64}\omega)f_{i-1}^k + (\tfrac{57}{128}+\tfrac{7}{64}\omega)f_i^k + (\tfrac{49}{128}+\tfrac{1}{64}\omega)f_{i+1}^k + (\tfrac{7}{128}-\tfrac{3}{64}\omega)f_{i+2}^k,$$

$$f_{4i+2}^{k+1} = (\tfrac{7}{128} - \tfrac{3}{64}\omega) f_{i-1}^{k} + (\tfrac{49}{128} + \tfrac{1}{64}\omega) f_{i}^{k} + (\tfrac{57}{128} + \tfrac{7}{64}\omega) f_{i+1}^{k} + (\tfrac{15}{128} - \tfrac{5}{64}\omega) f_{i+2}^{k} \,, \tag{3.3}$$

$$f_{4i+3}^{k+1} = (\tfrac{1}{64} - \tfrac{1}{64}\omega) f_{i-1}^{k} + (\tfrac{5}{16} - \tfrac{5}{64}\omega) f_{i}^{k} + (\tfrac{29}{64} + \tfrac{13}{64}\omega) f_{i+1}^{k} + (\tfrac{7}{32} - \tfrac{7}{64}\omega) f_{i+2}^{k} \,,$$

then scheme (3.3) is C^3-continuous over the interval $0 < \omega < 3/2$.

Corollary 3.5. Given $p_i^0 = p_i, i \in Z$, let the values p_i^k, $k \geq 1$ be defined recursively by the following 7-point interpolating ternary subdivision scheme of Lian (2009)

$$\begin{aligned} f_{3i}^{k+1} &= \tfrac{35}{6561} f_{i-3}^{k} - \tfrac{112}{2187} f_{i-2}^{k} + \tfrac{700}{2187} f_{i-1}^{k} + \tfrac{5600}{6561} f_{i}^{k} - \tfrac{350}{2187} f_{i+1}^{k} + \tfrac{80}{2187} f_{i+2}^{k} - \tfrac{28}{6561} f_{i+3}^{k} \,, \\ f_{3i+1}^{k+1} &= f_{i}^{k} \,, \\ f_{3i+2}^{k+1} &= -\tfrac{28}{6561} f_{i-3}^{k} + \tfrac{80}{2187} f_{i-2}^{k} - \tfrac{350}{2187} f_{i-1}^{k} + \tfrac{5600}{6561} f_{i}^{k} + \tfrac{700}{2187} f_{i+1}^{k} - \tfrac{112}{2187} f_{i+2}^{k} + \tfrac{35}{6561} f_{i+3}^{k} \end{aligned} \tag{3.4}$$

then scheme (3.4) is C^1-continuous.

Corollary 3.6.

Given $p_i^0 = p_i, i \in Z$, let the values p_i^k, $k \geq 1$ be defined recursively by following 6-point approximating quinary subdivision scheme of Mustafa and Rehman (2010)

$$\begin{aligned} f_{5i}^{k+1} &= \xi[18183 f_{i-2}^{k} - 173565 f_{i-1}^{k} + 3818430 f_{i}^{k} + 424270 f_{i+1}^{k} - 100485 f_{i+2}^{k} + 13167 f_{i+3}^{k}] \,, \\ f_{5i+1}^{k+1} &= \xi[41769 f_{i-2}^{k} - 369495 f_{i-1}^{k} + 3202290 f_{i}^{k} + 1372410 f_{i+1}^{k} - 282555 f_{i+2}^{k} + 35581 f_{i+3}^{k}] \,, \\ f_{5i+2}^{k+1} &= \xi[46875 f_{i-2}^{k} - 390625 f_{i-1}^{k} + 2343750 f_{i}^{k} + 2343750 f_{i+1}^{k} - 390625 f_{i+2}^{k} + 46875 f_{i+3}^{k}], \\ f_{5i+3}^{k+1} &= \xi[35581 f_{i-2}^{k} - 282555 f_{i-1}^{k} + 1372410 f_{i}^{k} + 3202290 f_{i+1}^{k} - 369495 f_{i+2}^{k} + 41769 f_{i+3}^{k}] \,, \\ f_{5i+4}^{k+1} &= \xi[13167 f_{i-2}^{k} - 100485 f_{i-1}^{k} + 424270 f_{i}^{k} + 3818430 f_{i+1}^{k} - 173565 f_{i+2}^{k} + 18183 f_{i+3}^{k}] \,, \end{aligned} \tag{3.5}$$

where $\xi = 1/4000000$ then scheme (3.5) is C^2-continuous.

3.4. Comparison

Here is a comparison between the proposed numerical algorithm (PNA) for continuity and the Laurent polynomial algorithm (LPA).

- We see that Step 1 of LPA and Step-1 of PNA are same.
- We also observe that Step 4(a & b) of LPA and Step 3 of PNA are same.
- Step 2 and Step 3 of LPA are different from Step 2 and Step-4 of PNA. Here we observe that polynomial factorization, division and summation are involved in Step 2 and Step 3 of LPA but simple arithmetic operations such as subtraction and multiplication are involved in Step 2 and Step 4 of PNA.
- Obviously, for higher arity schemes (like ternary, quaternary, etc.), the polynomial factorization, division and summation involved in Step 2 and Step 3 of LPA require more computations than simple arithmetic operations involved in Step 2 and Step 4 of PNA.
- However, for few schemes with negative masks without parameter, LPA gives sharp bounds for continuity than PNA. In this special case the PNA needs alternative of Step-4(c) of LPA to get sharp bound. We leave this as an open question.

4. Conclusion

In this paper, we have presented numerical algorithms for computing higher order divided differences and continuity of m-point n-ary subdivision schemes for m, $n \geq 2$. Our numerical algorithm for computing divided differences is relatively new. Proposed numerical algorithm for continuity is free from polynomial algebraic operations, numerically stable, fast and efficient. We have demonstrated the validity of the numerical algorithm by numerical examples. The results obtained by our numerical algorithm coincide with those of the Laurent polynomial algorithm.

Acknowledgement

This work is supported by the Indigenous Ph. D. Scholarship Scheme of Higher Education Commission (HEC) Pakistan.

REFERENCES

Aspert, N. (2003). Non-linear subdivision of univariate signals and discrete surfaces. Ph.D thesis, E'cole Polytechnique $Fe'de'$rale, de Lausanne, Lausanne, Switzerland.

Dyn, N. (2002). Analysis of convergence and smoothness by the formalism of Laurent polynomials. A. Iske, E. Quak, M. S. Floater (Eds), Tutorials on Multiresolution in Geometric Modeling, Springer, pp. 51-68 (Chapter 3).

Dyn, N., Levin, D. and J. A. Gregory, J. A. (1987). A 4-point interpolatory subdivision scheme for curve design. Computer Aided Geometric Design, Vol. 4, pp. 257-268.

Dyn, N., Gregory, J. A., and Levin, D. (1991). Analysis of uniform binary subdivision schemes for curve design. Constructive Approximation, pp. 127-147.

Sabin, M. (2010). Analysis and design of univariate subdivision schemes, Geometry and Computing, Springer 6, ISBN 978-3-642-13647-4.

Lian, J.-A. (2009). On *a*-ary subdivision for curve design: III. 2*m*-point and 2*m*+1-point interpolatory schemes, Applications and Applied Mathematics: An International Journal, Vol. 4, No. 2, pp. 434-444.

Mustafa, G. and Rehman, N. A. (2010). The mask of (2*b* + 4)-point *n*-ary subdivision scheme, Computing, Vol. 90, pp. 1-114. DOI 10.1007/s00607-010-0108-x.

Khan, F. and Mustafa, G. (2008). Ternary six-point interpolating subdivision scheme, Lobachevskii Journal of Mathematics, Vol. 3, pp. 153-163.

Hassan, M. F. and Dodgson, N. A. (2001). Ternary and three-point univariate subdivision schemes. TR. No. 520, 2001, University of Cambridge Computer Laboratory, http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-520.pdf.

Mustafa, G. and Khan, F. (2009). A new 4-point C^3 quaternary approximating subdivision scheme, Abstract and Applied Analysis, Vol. 2009, Article ID 301967, 14 pages, DOI:10.1155/2009/301967.

Zheng, H. Hu, M. and Peng, G. (2009). *p*-ary subdivision generalizing B-splines, Second International Conference on Computer and Electrical Engineering, DOI: 10.1109/ICCEE.2009.204.