

Article

# Viscovatov-Like Algorithm of Thiele–Newton’s Blending Expansion for a Bivariate Function

Shengfeng Li <sup>1,\*</sup>  and Yi Dong <sup>2</sup><sup>1</sup> Institute of Applied Mathematics, Bengbu University, Bengbu 233030, China<sup>2</sup> School of Science, Bengbu University, Bengbu 233030, China

\* Correspondence: lsf@bbc.edu.cn; Tel.: +86-552-317-5158

Received: 18 June 2019; Accepted: 30 July 2019; Published: 2 August 2019



**Abstract:** In this paper, Thiele–Newton’s blending expansion of a bivariate function is firstly suggested by means of combining Thiele’s continued fraction in one variable with Taylor’s polynomial expansion in another variable. Then, the Viscovatov-like algorithm is given for the computations of the coefficients of this rational expansion. Finally, a numerical experiment is presented to illustrate the practicability of the suggested algorithm. Henceforth, the Viscovatov-like algorithm has been considered as the imperative generalization to find out the coefficients of Thiele–Newton’s blending expansion of a bivariate function.

**Keywords:** bivariate function; divided difference; inverse difference; blending difference; continued fraction; Thiele–Newton’s expansion; Viscovatov-like algorithm

## 1. Introduction

The interpolation and expansion of a function are two of the oldest and most interesting branches in both computational mathematics and approximation theory. Most often, they have a natural link with their corresponding algorithms, such as Newton’s interpolatory formula and its divided-difference algorithm, Thiele’s interpolating continued fraction and its inverse-difference algorithm, Thiele’s expansion of a univariate function and its Viscovatov’s algorithm, and so on. For the function  $f$  being a univariate function, such problems have been extensively investigated, and abundant research results have been achieved. Some surveys and a complete literature for the problems in single variable interpolation and expansion can be found in Cheney [1], Hildebrand [2], Davis [3], Alfio et al. [4], Gautschi [5], Burden et al. [6], and the references therein. However, in comparison to the broad research and application of the univariate interpolation and expansion problems, much less attention has been paid to the problems associated with multivariate interpolation and expansion, and the study of multivariate rational interpolation and expansion is even less. However, fortunately, there exists some literature discussing the multivariate rational interpolation and expansion problems. We mention the works of Baker et al. [7,8], Kuchminskaya [9], Skorobogatko [10], Siemaszko [11], Viscovatov [12], Graves-Morris [13], Cuyt and Verdonk [14–17], Möller [18], Zhu et al. [19], Gu et al. [20,21], Tan et al. [22–28], and the references therein for results concerning the multivariate rational interpolation and expansion.

Skorobogatko applied the idea of the branch to the continued fraction from about the 1960s to the 1980s, which ushered in a new era of the research on the theories and methods of the continued fraction [10]. In 1983, the concept of the Thiele-type interpolation by the continued fraction in one variable was generalized to the multivariate case by Siemaszko [11], and the Thiele-type branched continued fractions were obtained and an algorithm for the computation of the limiting case of branched continued fractions for bivariable functions suggested. Furthermore, in the 1980s, based on the so-called symmetric branched continued fraction, Cuyt et al. [14–16] introduced a symmetric

interpolation scheme and studied the expansion of a bivariate function by using this method and technique. By following the prior works, in 1995, Zhu et al. [19,22] discussed the vector-valued rational interpolants by branched continued fractions. In 1997, Gu et al. [20,21] investigated the problem about matrix-valued rational interpolants. In the meantime, Tan et al. engaged in studying bivariate rational interpolants and obtained tremendous scholarly achievements in this field [22–28]. In 2007, Tan summarized the research results concerning the theory of the continued fraction and published the famous book *The Theory of Continued Fractions and Their Applications*. This book has played an important role in promoting some modern research about the continued fraction. Furthermore, there are a few works and references about the application of the continued fraction in image processing, such as the literature of Hu and Tan [29,30], Li et al. [31].

As we all know, Taylor’s expansion of a function is likely to be the best known and most widely-used formula for the function approximation problem. If  $f$  is a function of a univariate  $x$  and the derivatives of all orders are uniformly bounded in a neighborhood  $\mathcal{U}(\xi)$ , then for each  $x$  in  $\mathcal{U}(\xi)$ ,  $f(x)$  can be expanded into the following Taylor’s formula about  $\xi$ :

$$f(x) = C_0 + C_1(x - \xi) + C_2(x - \xi)^2 + \dots + C_k(x - \xi)^k + \dots,$$

where  $C_k = \frac{1}{k!} f^{(k)}(\xi), k = 0, 1, 2, \dots$ . On the other hand, the function  $f(x)$  can also be expanded about  $\xi$  in terms of the continued fraction, which is in the form of the following:

$$f(x) = d_0 + \cfrac{x - \xi}{d_1} + \cfrac{x - \xi}{d_2} + \dots + \cfrac{x - \xi}{d_n} + \dots,$$

where  $d_k \in \mathfrak{R}, k = 0, 1, 2, \dots$ . Here, the above formula is called Thiele’s expansion for  $f(x)$  about  $\xi$ . There is a famous algorithm to compute the coefficients  $d_0, d_1, d_2, \dots$ , of Thiele’s expansion, which is called Viscovatov’s algorithm. We can see the references [16,28].

Motivated by the results concerning the univariate function, in this paper, we consider the rational expansion by Thiele’s continued fraction of a bivariate function and give a Viscovatov-like algorithm for the computations of the coefficients. As a preliminary to our discussions, Thiele–Newton’s interpolation needs to be introduced first. In the works [25,28], the so-called Thiele–Newton’s interpolation was suggested to construct bivariate interpolants by Tan et al. Its main idea is to combine Thiele’s interpolating continued fraction in one variable with Newton’s interpolating polynomial in another variable to hybridize a new interpolation, which is defined as below:

$$TN_{m,n}(x, y) = t_0(y) + \cfrac{x - x_0}{t_1(y)} + \cfrac{x - x_1}{t_2(y)} + \dots + \cfrac{x - x_{m-1}}{t_m(y)}, \tag{1}$$

where:

$$t_i(y) = \varphi_{TN}[x_0, \dots, x_i; y_0] + (y - y_0)\varphi_{TN}[x_0, \dots, x_i; y_0, y_1] + \dots + (y - y_0)(y - y_1) \dots (y - y_{n-1})\varphi_{TN}[x_0, \dots, x_i; y_0, \dots, y_n] \tag{2}$$

for  $i = 0, 1, \dots, m$ , both  $X = \{x_i | i \in \mathbb{N}\}$  and  $Y = \{y_j | j \in \mathbb{N}\}$  are two sets of points belonging to  $\mathfrak{R}$ , and  $\varphi_{TN}[x_0, \dots, x_i; y_0, \dots, y_j]$  denotes the blending difference of the function  $f(x, y)$  at points  $x_0, \dots, x_i; y_0, \dots, y_j$ . Suppose that any blending difference  $\varphi_{TN}[x_0, \dots, x_i; y_0, \dots, y_j]$  exists. Then, one can easily confirm that:

$$TN_{m,n}(x_i, y_j) = f(x_i, y_j), \quad i = 0, 1, \dots, m; \quad j = 0, 1, \dots, n.$$

The limiting case of Thiele’s interpolating continued fraction expansion of a univariate function has been discussed in the literature [26]. With the inspiration of the limiting case, Thiele–Newton’s expansion of a bivariate function is yielded when all the points in sets  $X = \{x_i | i \in \mathbb{N}\}$  and  $Y = \{y_j | j \in$

$\mathbb{N}$  are coincident with certain points  $\xi$  and  $\zeta$ , respectively, from Equations (1) and (2), or in other words, a bivariate function  $f(x, y)$  has Thiele–Newton’s expansion of the following form:

$$f(x, y) = l_0(y) + \frac{x - \xi}{l_1(y)} + \frac{x - \xi}{l_2(y)} + \dots + \frac{x - \xi}{l_m(y)}, \tag{3}$$

where:

$$l_i(y) = a_{i,0} + a_{i,1}(y - \zeta) + a_{i,2}(y - \zeta)^2 + a_{i,3}(y - \zeta)^3 + \dots \tag{4}$$

for all  $i \in \mathbb{N}$ . Therefore, there exists a question about how to calculate the unknowns  $a_{i,j}, i = 0, 1, 2, \dots, j = 0, 1, 2, \dots$ , in Equation (4).

The aim of this paper is to find an algorithm for the computations of the coefficients of Thiele–Newton’s expansion of a bivariate function. The paper is organized as follows. In Section 2, we briefly recall some preliminaries for Thiele’s continued fraction and Thiele–Newton’s blending interpolation. In Section 3, we suggest Thiele–Newton’s blending rational expansion and prove the Viscovatov-like algorithm. In Section 4, numerical examples are given to illustrate the application of the Viscovatov-like algorithm. Throughout the paper, we let  $\mathbb{N}$  and  $\mathfrak{R}$  stand for the set of natural numbers and the set of real numbers, respectively.

### 2. Preliminaries

In this section, we briefly review some basic definitions and results for Thiele’s continued fraction, Thiele’s expansion of a univariate function, and blending interpolation. Some surveys and complete literature about the continued fraction could be found in Cuyt et al. [14–16], Zhu et al. [19], Gu et al. [20,21], and Tan et al. [25,26,28].

**Definition 1.** Assume that  $G$  is a subset of the complex plane and  $X = \{x_i | i \in \mathbb{N}\}$  is a set of points belonging to  $G$ . Suppose, in addition, that  $f(x)$  is a function defined on  $G$ . Let:

$$\begin{aligned} f[x_i] &= f(x_i), i \in \mathbb{N}, \\ f[x_i, x_j] &= \frac{f[x_i] - f[x_j]}{x_i - x_j}, \\ f[x_i, x_j, x_k] &= \frac{f[x_i, x_k] - f[x_i, x_j]}{x_k - x_j} \end{aligned}$$

and:

$$f[x_i, \dots, x_j, x_k, x_l] = \frac{f[x_i, \dots, x_j, x_l] - f[x_i, \dots, x_j, x_k]}{x_l - x_k}.$$

Then,  $f[x_i, \dots, x_j, x_k]$  is called the divided difference of  $f(x)$  with respect to points  $x_i, \dots, x_j, x_k$ .

**Definition 2.** Assume that  $G$  is a subset of the complex plane and  $X = \{x_i | i \in \mathbb{N}\}$  is a set of points in  $G$ . Suppose, in addition, that  $f(x)$  is a function defined on  $G$ . We let:

$$\begin{aligned} \rho[x_i] &= f(x_i), i \in \mathbb{N}, \\ \rho[x_i, x_j] &= \frac{x_i - x_j}{\rho[x_i] - \rho[x_j]}, \\ \rho[x_i, x_j, x_k] &= \frac{x_k - x_j}{\rho[x_i, x_k] - \rho[x_i, x_j]} \end{aligned}$$

and:

$$\rho[x_i, \dots, x_j, x_k, x_l] = \frac{x_l - x_k}{\rho[x_i, \dots, x_j, x_l] - \rho[x_i, \dots, x_j, x_k]}.$$

Then,  $\rho[x_i, \dots, x_j, x_k]$  is called the inverse difference of  $f(x)$  with respect to points  $x_i, \dots, x_j, x_k$ .

**Definition 3.** Assume that  $G$  is a subset of the complex plane and  $X = \{x_i | i \in \mathbb{N}\} \subseteq G$  is a set of points. In addition, let  $f(x)$  be a function defined on  $G$ , and let:

$$R_n(x) = \rho[x_0] + \frac{x - x_0}{\rho[x_0, x_1]} + \dots + \frac{x - x_{n-1}}{\rho[x_0, x_1, \dots, x_n]}, \tag{5}$$

where  $\rho[x_0, x_1, \dots, x_i], i = 0, 1, 2, \dots, n$ , is the inverse difference of  $f(x)$  with respect to points  $x_0, x_1, \dots, x_i$ . Then,  $R_n(x)$  is called Thiele’s interpolating continued fraction of order  $n$ . It is easy to verify that the rational function satisfies the following conditions:

$$R_n(x_i) = f(x_i), i = 0, 1, 2, \dots, n.$$

When all the points in the set  $X = \{x_i | i \in \mathbb{N}\}$  are coincident with a certain point  $\xi \in G$ , Thiele’s expansion of a univariate function  $f(x)$  at  $x = \xi$  is obtained as follows:

$$f(x) = d_0 + \frac{x - \xi}{d_1} + \frac{x - \xi}{d_2} + \dots + \frac{x - \xi}{d_n} + \dots, \tag{6}$$

where  $d_k \in \mathfrak{R}, k = 0, 1, 2, \dots$ . Moreover, if  $f(x)$  is a function with derivatives of all orders in a neighborhood  $\mathcal{U}(\xi)$ , then Taylor’s expansion of the function  $f(x)$  at  $x = \xi$  is denoted as below:

$$f(x) = \sum_{n=0}^{\infty} C_n^{(0)}(x - \xi)^n,$$

where  $C_n^{(0)} = \frac{1}{n!} f^{(n)}(\xi), n = 0, 1, 2, \dots$ . A famous method, called Viscovatov’s algorithm (see [16,28]), is available for the computations of the coefficients  $d_0, d_1, d_2, \dots$ , of Thiele’s expansion, which is formulated as follows.

**Algorithm 1.** Viscovatov’s algorithm to calculate the coefficients  $d_0, d_1, d_2, \dots$ :

$$\begin{aligned} C_i^{(0)} &= f^{(i)}(\xi) / i!, i = 0, 1, 2, \dots, \\ d_0 &= C_0^{(0)}, \\ d_1 &= 1 / C_1^{(0)}, \\ C_i^{(1)} &= -C_{i+1}^{(0)} / C_1^{(0)}, i \geq 1, \\ d_l &= C_1^{(l-2)} / C_1^{(l-1)}, l \geq 2, \\ C_i^{(l)} &= C_{i+1}^{(l-2)} - d_l C_{i+1}^{(l-1)}, i \geq 1, l \geq 2. \end{aligned}$$

**Remark 1.** Clearly, by applying Viscovatov’s algorithm, we can carry out computations step by step for the coefficients  $d_0, d_1, d_2, \dots$ .

In [25,28], the method known as Thiele–Newton’s blending interpolation was suggested to construct bivariate interpolants by Tan et al. Before the method can be introduced, we recall the definition concerning the blending difference.

**Definition 4.** Assume that  $\Pi_{m,n} = X_m \times Y_n$ , where  $X_m = \{x_i | i = 0, 1, 2, \dots, m\} \subset [a, b] \subset \mathfrak{R}$  and  $Y_n = \{y_j | j = 0, 1, 2, \dots, n\} \subset [c, d] \subset \mathfrak{R}$  are two sets of points. Suppose that  $f(x, y)$  is a function of two variables defined on  $D = [a, b] \times [c, d]$ . Let:

$$\begin{aligned} \varphi_{TN}[x_i; y_j] &= f(x_i, y_j), (x_i, y_j) \in D, \\ \varphi_{TN}[x_i; y_p, y_q] &= \frac{\varphi_{TN}[x_i; y_q] - \varphi_{TN}[x_i; y_p]}{y_q - y_p}, \\ \varphi_{TN}[x_i; y_p, \dots, y_q, y_r, y_s] &= \frac{\varphi_{TN}[x_i; y_p, \dots, y_q, y_s] - \varphi_{TN}[x_i; y_p, \dots, y_q, y_r]}{y_s - y_r}, \\ \varphi_{TN}[x_i, x_j; y_p] &= \frac{x_j - x_i}{\varphi_{TN}[x_j; y_p] - \varphi_{TN}[x_i; y_p]}, \\ \varphi_{TN}[x_i, \dots, x_j, x_k, x_l; y_p] &= \frac{x_l - x_k}{\varphi_{TN}[x_i, \dots, x_j, x_l; y_p] - \varphi_{TN}[x_i, \dots, x_j, x_k; y_p]} \end{aligned}$$

and:

$$\varphi_{TN}[x_i, \dots, x_l; y_p, \dots, y_q, y_r, y_s] = \frac{\varphi_{TN}[x_i, \dots, x_l; y_p, \dots, y_q, y_s] - \varphi_{TN}[x_i, \dots, x_l; y_p, \dots, y_q, y_r]}{y_s - y_r}.$$

Then,  $\varphi_{TN}[x_0, \dots, x_i; y_0, \dots, y_j]$  is called Thiele–Newton’s blending difference of  $f(x, y)$  with respect to the set of points  $\Pi_{i,j}$ .

**Remark 2.** From Definition 4, it is easy to see that the first recurrence relations on Thiele–Newton’s blending difference  $\varphi_{TN}[x_0, \dots, x_i; y_0, \dots, y_j]$  are just the inverse difference of  $f(x, y)$  with respect to the variable  $x$ , and the second recurrence relations are only the divided difference of  $f(x, y)$  with respect to the variable  $y$ .

Next, recall Thiele–Newton’s interpolation  $TN_{m,n}(x, y)$ , as shown in Equations (1) and (2). In order to calculate this rational interpolation, we need to utilize the following algorithm whose main operation is matrix transformations (see [23,28]).

Algorithm 2. Four main steps for the algorithm to calculate Thiele–Newton’s interpolation are as follows:

- **Step 1: Initialization.** For  $i = 0, 1, \dots, m$ ;  $j = 0, 1, \dots, n$ , let  $f_{i,j}^{(0,0)} = f(x_i, y_j)$ . Define the following initial information matrix:

$$M_0 = \begin{bmatrix} f_{0,0}^{(0,0)} & f_{1,0}^{(0,0)} & \cdots & f_{m,0}^{(0,0)} \\ f_{0,1}^{(0,0)} & f_{1,1}^{(0,0)} & \cdots & f_{m,1}^{(0,0)} \\ \vdots & \vdots & \ddots & \vdots \\ f_{0,n}^{(0,0)} & f_{1,n}^{(0,0)} & \cdots & f_{m,n}^{(0,0)} \end{bmatrix}.$$

- **Step 2: Thiele’s recursion along the X-axis.** For  $j = 0, 1, \dots, n$ ;  $p = 1, 2, \dots, m$ ;  $i = p, p + 1, \dots, m$ , compute:

$$f_{i,j}^{(p,0)} = \frac{x_i - x_{p-1}}{f_{i,j}^{(p-1,0)} - f_{p-1,j}^{(p-1,0)}}$$

and construct the following information matrix:

$$M_1 = \begin{bmatrix} f_{0,0}^{(0,0)} & f_{1,0}^{(1,0)} & \dots & f_{m,0}^{(m,0)} \\ f_{0,1}^{(0,0)} & f_{1,1}^{(1,0)} & \dots & f_{m,1}^{(m,0)} \\ \vdots & \vdots & \ddots & \vdots \\ f_{0,n}^{(0,0)} & f_{1,n}^{(1,0)} & \dots & f_{m,n}^{(m,0)} \end{bmatrix}.$$

- **Step 3: Newton’s recursion along the Y-axis.** For  $i = 0, 1, \dots, m; q = 1, 2, \dots, n; j = q, q + 1, \dots, n$ , compute:

$$f_{i,j}^{(i,q)} = \frac{f_{i,j}^{(i,q-1)} - f_{i,q-1}^{(i,q-1)}}{y_j - y_{q-1}}$$

and construct the following information matrix:

$$M_2 = \begin{bmatrix} f_{0,0}^{(0,0)} & f_{1,0}^{(1,0)} & \dots & f_{m,0}^{(m,0)} \\ f_{0,1}^{(0,1)} & f_{1,1}^{(1,1)} & \dots & f_{m,1}^{(m,1)} \\ \vdots & \vdots & \ddots & \vdots \\ f_{0,n}^{(0,n)} & f_{1,n}^{(1,n)} & \dots & f_{m,n}^{(m,n)} \end{bmatrix}.$$

- **Step 4: Establish Thiele–Newton’s interpolation.** For  $i = 0, 1, \dots, m$ , let:

$$t_{i,n}(y) = f_{i,0}^{(i,0)} + (y - y_0)f_{i,1}^{(i,1)} + \dots + (y - y_0)(y - y_1) \dots (y - y_{n-1})f_{i,n}^{(i,n)}.$$

Then, Thiele–Newton’s interpolation is established as follows:

$$TN_{m,n}(x, y) = t_{0,n}(y) + \frac{x - x_0}{t_{1,n}(y)} + \frac{x - x_1}{t_{2,n}(y)} + \dots + \frac{x - x_{m-1}}{t_{m,n}(y)},$$

which satisfies:

$$TN_{m,n}(x_i, y_j) = f(x_i, y_j)$$

for  $i = 0, 1, \dots, m; j = 0, 1, \dots, n$ .

**Remark 3.** Obviously, for any  $i \in \{0, 1, \dots, m\}$ , by using the elements  $f_{i,j}^{(i,j)}, j = 0, 1, \dots, n$ , in the  $(i + 1)$ th column of the information matrix  $M_2$ , Newton’s interpolating polynomial  $t_{i,n}(y)$  with respect to the variable  $y$  can be constructed.

### 3. Thiele–Newton’s Blending Expansion and the Viscovatov-Like Algorithm

In this section, our main objective is to expound on Thiele–Newton’s blending rational expansion of a bivariate function and show the Viscovatov-like algorithm that finds the coefficients of Thiele–Newton’s expansion.

#### 3.1. Thiele–Newton’s Blending Expansion

**Definition 5.** Assume that  $\Pi = X \times Y$  with  $\Pi \subset D = [a, b] \times [c, d]$ , where  $X = \{x_i | i = 0, 1, 2, \dots\} \subset [a, b] \subset \mathfrak{R}$  and  $Y = \{y_j | j = 0, 1, 2, \dots\} \subset [c, d] \subset \mathfrak{R}$  are two sets of points. Suppose, in addition, that the point  $(\xi, \zeta) \in D$  and  $f(x, y)$  is a bivariate function defined on  $D$ . Let all the points in the set  $X = \{x_i | i = 0, 1, 2, \dots\}$  and  $Y = \{y_j | j = 0, 1, 2, \dots\}$  be coincident with the given points  $\xi$  and  $\zeta$ , respectively. Then, Thiele–Newton’s

interpolation  $TN_{m,n}(x, y)$  of a bivariate function  $f(x, y)$  defined in Section 2 turns into Thiele–Newton’s blending expansion of the bivariate function  $f(x, y)$  as shown below:

$$f(x, y) = d_0(y) + \frac{x - \xi}{d_1(y)} + \frac{x - \xi}{d_2(y)} + \frac{x - \xi}{d_3(y)} + \dots, \tag{7}$$

where:

$$d_i(y) = a_{i,0} + a_{i,1}(y - \zeta) + a_{i,2}(y - \zeta)^2 + a_{i,3}(y - \zeta)^3 + \dots \tag{8}$$

for any  $i \in \mathbb{N}$ .

Obviously, a main topic for further discussion is how to calculate the coefficients  $d_i(y), i = 0, 1, 2, \dots$ , in Equation (7), or in other words, how to compute the coefficients  $a_{i,j}, i = 0, 1, 2, \dots; j = 0, 1, 2, \dots$ , in Equation (8). In order to handle the problem that we are facing in the bivariate case, we introduce the following algorithm.

### 3.2. Viscovatov-Like Algorithm

Suppose that  $f(x, y)$  is a bivariate function of two variables  $x$  and  $y$ . If  $y$  is held constant, say  $y = \zeta$ , then  $f(x, \zeta)$  is a function of the single variable  $x$ . Likewise,  $f(\xi, y)$  is also a function of the single variable  $y$  when  $x$  is regarded as a constant, i.e.,  $x = \xi$ . We use the notation:  $D_x^m f(x, y)$  denotes the  $m$ -order partial derivative of  $f(x, y)$  with respect to  $x$ . Similarly, the  $n$ -order partial derivative of  $f(x, y)$  with respect to  $y$  is denoted by  $D_y^n f(x, y)$ . Furthermore,  $D_x^m f(\xi, \zeta)$  and  $D_y^n f(\xi, \zeta)$  denote the values of  $D_x^m f(x, y)$  and  $D_y^n f(x, y)$  about the point  $(x, y) = (\xi, \zeta)$ , respectively. Let:

$$C_k^{(0)}(y) = \frac{1}{k!} D_x^k f(\xi, y), k = 0, 1, 2, \dots$$

Then, the bivariate function  $f(x, y)$  can be expanded formally about the point  $\xi$  as follows:

$$f(x, y) = C_0^{(0)}(y) + C_1^{(0)}(y)(x - \xi) + \dots + C_k^{(0)}(y)(x - \xi)^k + \dots \tag{9}$$

From Equations (7)–(9), we give the Viscovatov-like algorithm, which finds out the coefficients of Thiele–Newton’s expansion  $d_i(y), i = 0, 1, 2, \dots$ , and  $a_{i,j}, i = 0, 1, 2, \dots; j = 0, 1, 2, \dots$ , as described by the following algorithm.

**Algorithm 3.** Viscovatov-like algorithm to calculate the coefficients  $d_i(y), i = 0, 1, 2, \dots$ , and  $a_{i,j}, i = 0, 1, 2, \dots; j = 0, 1, 2, \dots$ :

$$\begin{aligned} C_i^{(0)}(y) &= D_x^i f(\xi, y) / i!, i = 0, 1, 2, \dots, \\ d_0(y) &= C_0^{(0)}(y) = f(\xi, y), \\ d_1(y) &= 1 / C_1^{(0)}(y), \\ C_i^{(1)}(y) &= -C_{i+1}^{(0)}(y) / C_1^{(0)}(y), i \geq 1, \\ d_l(y) &= C_1^{(l-2)}(y) / C_1^{(l-1)}(y), l \geq 2, \\ C_i^{(l)}(y) &= C_{i+1}^{(l-2)}(y) - d_l(y) C_{i+1}^{(l-1)}(y), i \geq 1, l \geq 2, \\ a_{i,j} &= D_y^j d_i(\zeta) / j!, i = 0, 1, 2, \dots; j = 0, 1, 2, \dots \end{aligned}$$

**Proof of Algorithm 3.** First, we compute the coefficients  $d_0(y)$  and  $d_1(y)$ . Considering the two expansions (7) and (9), we have:

$$C_0^{(0)}(y) + C_1^{(0)}(y)(x - \xi) + C_2^{(0)}(y)(x - \xi)^2 + \dots = d_0(y) + \frac{x - \xi}{d_1(y)} + \frac{x - \xi}{d_2(y)} + \frac{x - \xi}{d_3(y)} + \dots \tag{10}$$

Letting  $x = \zeta$ , from Equation (10), one can clearly get:

$$d_0(y) = C_0^{(0)}(y). \tag{11}$$

Combining Equation (11) with Equation (10), we have:

$$d_1(y) + \frac{x - \zeta}{|d_2(y)|} + \frac{x - \zeta}{|d_3(y)|} + \dots = \frac{1}{C_1^{(0)}(y) + C_2^{(0)}(y)(x - \zeta) + C_3^{(0)}(y)(x - \zeta)^2 + \dots}. \tag{12}$$

Let  $x = \zeta$  in Equation (12). Then, we can easily obtain:

$$d_1(y) = \frac{1}{C_1^{(0)}(y)}. \tag{13}$$

Next, by mathematical induction, we shall prove that the following equation:

$$d_l(y) = \frac{C_1^{(l-2)}(y)}{C_1^{(l-1)}(y)} \tag{14}$$

is true for all  $l \geq 2$ .

When  $l = 2$ , we shall verify that Equation (14) holds. Substituting Equation (13) into Equation (12), we have:

$$\begin{aligned} \frac{x - \zeta}{|d_2(y)|} + \frac{x - \zeta}{|d_3(y)|} + \dots &= \frac{1}{C_1^{(0)}(y) \left[ 1 + \frac{C_2^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta) + \frac{C_3^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta)^2 + \dots \right]} - \frac{1}{C_1^{(0)}(y)} \\ &= - \frac{\frac{C_2^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta) + \frac{C_3^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta)^2 + \dots}{C_1^{(0)}(y) + C_2^{(0)}(y)(x - \zeta) + C_3^{(0)}(y)(x - \zeta)^2 + \dots}, \end{aligned} \tag{15}$$

which implies that:

$$d_2(y) + \frac{x - \zeta}{|d_3(y)|} + \frac{x - \zeta}{|d_4(y)|} + \dots = \frac{C_1^{(0)}(y) + C_2^{(0)}(y)(x - \zeta) + C_3^{(0)}(y)(x - \zeta)^2 + \dots}{- \frac{C_2^{(0)}(y)}{C_1^{(0)}(y)} - \frac{C_3^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta) - \frac{C_4^{(0)}(y)}{C_1^{(0)}(y)}(x - \zeta)^2 - \dots}. \tag{16}$$

Let:

$$C_i^{(1)}(y) = - \frac{C_{i+1}^{(0)}(y)}{C_1^{(0)}(y)}, i = 1, 2, 3, \dots \tag{17}$$

Then, it follows from the identity (16) that:

$$d_2(y) + \frac{x - \zeta}{|d_3(y)|} + \frac{x - \zeta}{|d_4(y)|} + \dots = \frac{C_1^{(0)}(y) + C_2^{(0)}(y)(x - \zeta) + C_3^{(0)}(y)(x - \zeta)^2 + \dots}{C_1^{(1)}(y) + C_2^{(1)}(y)(x - \zeta) + C_3^{(1)}(y)(x - \zeta)^2 + \dots}. \tag{18}$$

Using  $x = \zeta$  in Equation (18) yields:

$$d_2(y) = \frac{C_1^{(0)}(y)}{C_1^{(1)}(y)}, \tag{19}$$

which implies that Equation (14) is true for  $l = 2$ .

When  $l \geq 3$ , assume that Equation (14) holds for any  $l = n, n = 3, 4, \dots$ . Then, let us prove that Equation (14) is also true for  $l = n + 1$ .

By assumption, we have the following equation:

$$d_n(y) = \frac{C_1^{(n-2)}(y)}{C_1^{(n-1)}(y)} \tag{20}$$

holds.

Referring to Equation (18), we assume that the following equation:

$$d_n(y) + \left| \frac{x - \xi}{d_{n+1}(y)} \right| + \left| \frac{x - \xi}{d_{n+2}(y)} \right| + \dots = \frac{C_1^{(n-2)}(y) + C_2^{(n-2)}(y)(x - \xi) + C_3^{(n-2)}(y)(x - \xi)^2 + \dots}{C_1^{(n-1)}(y) + C_2^{(n-1)}(y)(x - \xi) + C_3^{(n-1)}(y)(x - \xi)^2 + \dots} \tag{21}$$

is true, where:

$$C_i^{(k)}(y) = C_{i+1}^{(k-1)}(y) - d_k(y)C_{i+1}^{(k-1)}(y), k = n - 2, n - 1; n \geq 2; i = 1, 2, 3, \dots \tag{22}$$

Combining Equation (20) with Equation (21), one has:

$$\begin{aligned} & \left| \frac{x - \xi}{d_{n+1}(y)} \right| + \left| \frac{x - \xi}{d_{n+2}(y)} \right| + \dots \\ &= \frac{C_1^{(n-2)}(y) + C_2^{(n-2)}(y)(x - \xi) + C_3^{(n-2)}(y)(x - \xi)^2 + \dots}{C_1^{(n-1)}(y) \left[ 1 + \frac{C_2^{(n-1)}(y)}{C_1^{(n-1)}(y)}(x - \xi) + \frac{C_3^{(n-1)}(y)}{C_1^{(n-1)}(y)}(x - \xi)^2 + \dots \right]} - \frac{C_1^{(n-2)}(y)}{C_1^{(n-1)}(y)} \\ &= \frac{\left( C_2^{(n-2)}(y) - d_n(y)C_2^{(n-1)}(y) \right) (x - \xi) + \left( C_3^{(n-2)}(y) - d_n(y)C_3^{(n-1)}(y) \right) (x - \xi)^2 + \dots}{C_1^{(n-1)}(y) + C_2^{(n-1)}(y)(x - \xi) + C_3^{(n-1)}(y)(x - \xi)^2 + \dots} \end{aligned} \tag{23}$$

Let:

$$C_i^{(n)}(y) = C_{i+1}^{(n-2)}(y) - d_n(y)C_{i+1}^{(n-1)}(y), i = 1, 2, 3, \dots \tag{24}$$

Then, Equation (23) is rewritten as follows:

$$d_{n+1}(y) + \left| \frac{x - \xi}{d_{n+2}(y)} \right| + \left| \frac{x - \xi}{d_{n+3}(y)} \right| + \dots = \frac{C_1^{(n-1)}(y) + C_2^{(n-1)}(y)(x - \xi) + C_3^{(n-1)}(y)(x - \xi)^2 + \dots}{C_1^{(n)}(y) + C_2^{(n)}(y)(x - \xi) + C_3^{(n)}(y)(x - \xi)^2 + \dots}, \tag{25}$$

Using the above Equation (25) with  $x = \xi$  produces:

$$d_{n+1}(y) = \frac{C_1^{(n-1)}(y)}{C_1^{(n)}(y)}, \tag{26}$$

which means that Equation (14) holds for  $l = n + 1$ .

As is shown above, Equation (14) is true by mathematical induction for all  $l \geq 2$ . Meanwhile, we show that Equation (24) is also true for any  $l = n, n \geq 2$ .

Moreover, by differentiating Equation (8)  $j$  times with respect to the variable  $y$ , one has:

$$D_y^j d_i(y) = a_{i,j}j! + a_{i,j+1} \frac{(j+1)!}{1!} (y - \xi) + a_{i,j+2} \frac{(j+2)!}{2!} (y - \xi)^2 + \dots \tag{27}$$

Notice  $y = \zeta$ , from Equation (27), and we immediately obtain:

$$a_{i,j} = \frac{D_y^j a_i(\zeta)}{j!} \tag{28}$$

for  $i \in \mathbb{N}$  and  $j \in \mathbb{N}$ .

Therefore, associating Equation (28) with Equations (11), (13), (14), (17), and (24), we have shown the desired conclusion denoted by Algorithm 3. This completes the proof.  $\square$

### 4. Numerical Experiments

In the section, we give the results of numerical experiments to compare the efficiency of Thiele–Newton’s blending expansion (7) with series expansion of bivariate functions.

For  $|x| < 1, |y| < 1$  and  $x \neq y$ , given the following two test functions:

$$f_1(x, y) = \frac{1}{y - x} [\ln(1 - x) - \ln(1 - y)] \tag{29}$$

and:

$$f_2(x, y) = \frac{x^2}{(1 - x)(x - y)^2} + \frac{y^2}{(1 - y)(x - y)^2} + \frac{2xy [\ln(1 - x) - \ln(1 - y)]}{(x - y)^3}, \tag{30}$$

where  $\ln(z)$  gives the natural logarithm of  $z$  (logarithm to base  $e$ ). We shall discuss Thiele–Newton’s blending expansions of Equations (29) and (30), respectively.

First of all, let us consider Thiele–Newton’s blending expansion of the bivariate function  $f_1(x, y)$  defined by Equation (29) at the point  $(\xi, \zeta) = (0, 0)$ . Therefore, using the Viscovatov-like algorithm, we can obtain the coefficient using the notation  $a_{i,j}^{f_1}$  of Thiele–Newton’s expansion of  $f_1(x, y)$ . Some values of  $a_{i,j}^{f_1}, i = 0, 1, 2, \dots, m, \dots; j = 0, 1, 2, \dots, n, \dots$ , are shown in Table 1.

**Table 1.** The coefficients  $a_{i,j}^{f_1}$  of Thiele–Newton’s expansion of  $f_1(x, y)$  given by Equation (29).

$a_{i,j}^{f_1}$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$\dots$
$i = 0$	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\dots$
$i = 1$	2	$-\frac{4}{3}$	$-\frac{1}{9}$	$-\frac{8}{135}$	$-\frac{31}{810}$	$\dots$
$i = 2$	$-\frac{3}{4}$	$-\frac{7}{16}$	$-\frac{293}{960}$	$-\frac{299}{1280}$	$-\frac{33869}{179200}$	$\dots$
$i = 3$	16	$-\frac{88}{15}$	$-\frac{191}{225}$	$-\frac{10264}{23625}$	$-\frac{194491}{708750}$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

Thus, Thiele–Newton’s blending expansion of  $f_1(x, y)$  at  $(\xi, \zeta) = (0, 0)$  is denoted in the form:

$$\begin{aligned}
 f_1(x, y) = R^{f_1}(x, y) = & 1 + \frac{1}{2}y + \frac{1}{3}y^2 + \frac{1}{4}y^3 + \frac{1}{5}y^4 + \dots \\
 & + \frac{x}{\left[ 2 - \frac{4}{3}y - \frac{1}{9}y^2 - \frac{8}{135}y^3 - \frac{31}{810}y^4 + \dots \right]} \\
 & + \frac{x}{\left[ -\frac{3}{4} - \frac{7}{16}y - \frac{293}{960}y^2 - \frac{299}{1280}y^3 - \frac{33869}{179200}y^4 + \dots \right]} \\
 & + \frac{x}{\left[ 16 - \frac{88}{15}y - \frac{191}{225}y^2 - \frac{10264}{23625}y^3 - \frac{194491}{708750}y^4 + \dots \right]} + \dots \tag{31}
 \end{aligned}$$



Therefore, according to the values of  $a_{i,j}^{f_2}, i = 0, 1, 2, \dots; j = 0, 1, 2, \dots$ , in Table 2, the Thiele–Newton’s blending expansion of  $f_2(x, y)$  at  $(\xi, \zeta) = (0, 0)$  can be written as:

$$\begin{aligned}
 f_2(x, y) = R^{f_2}(x, y) = & 1 + y + y^2 + y^3 + y^4 + \dots \\
 & + \left\lfloor \frac{x}{1 - \frac{4}{3}y + \frac{5}{18}y^2 + \frac{4}{135}y^3 + \frac{17}{1620}y^4 + \dots} \right\rfloor \\
 & + \left\lfloor \frac{x}{-1 - \frac{7}{6}y - \frac{221}{180}y^2 - \frac{151}{120}y^3 - \frac{10721}{8400}y^4 + \dots} \right\rfloor + \dots .
 \end{aligned} \tag{38}$$

The corresponding truncated Thiele–Newton’s blending expansion  $R_{2,3}^{f_2}(x, y)$  of  $R^{f_2}(x, y)$  is:

$$R_{2,3}^{f_2}(x, y) = 1 + y + y^2 + y^3 + \frac{x}{1 - \frac{4}{3}y + \frac{5}{18}y^2 + \frac{4}{135}y^3 - \frac{\frac{x}{1 + \frac{7}{6}y + \frac{221}{180}y^2 + \frac{151}{120}y^3}}{}} . \tag{39}$$

By a similar technique, consider the Appell series for the bivariate function  $f_2(x, y)$  expanded about the point  $(\xi, \zeta) = (0, 0)$ ,

$$F1^{f_2}(1, 2, 2; 2; x, y) = \sum_{i,j=0}^{\infty} \frac{(1)_{i+j}(2)_i(2)_j}{(2)_{i+j}i!j!} x^i y^j = \sum_{i,j=0}^{\infty} \frac{(i+1)(j+1)}{i+j+1} x^i y^j . \tag{40}$$

The (2, 3)-order truncated Appell series for  $f_2(x, y)$  is:

$$F1_{2,3}^{f_2}(1, 2, 2; 2; x, y) = 1 + x + x^2 + y + \frac{4}{3}xy + \frac{3}{2}x^2y + y^2 + \frac{3}{2}xy^2 + \frac{9}{5}x^2y^2 + y^3 + \frac{8}{5}xy^3 + 2x^2y^3 . \tag{41}$$

Considering the errors, we let:

$$e_{2,3}^{f_k} = f_k(x, y) - R_{2,3}^{f_k}(x, y) \tag{42}$$

and:

$$E_{2,3}^{f_k} = f_k(x, y) - F1_{2,3}^{f_k}(a, b, c; d; x, y) \tag{43}$$

for  $k = 1, 2$ .

Table 3 lists various values of  $(x, y)$ , together with the values of the bivariate function  $f_1(x, y)$ , the truncated Thiele–Newton’s blending expansion  $R_{2,3}^{f_1}(x, y)$ , and the truncated Appell series  $F1_{2,3}^{f_1}(1, 1, 1; 2; x, y)$ . Furthermore, for comparison purposes, the values of errors  $e_{2,3}^{f_1}$  and  $E_{2,3}^{f_1}$  are given in this table. It can be seen from Table 3 that the error  $e_{2,3}^{f_1}$  using the truncated Thiele–Newton’s blending expansion  $R_{2,3}^{f_1}(x, y)$  is less than when using the truncated Appell series  $F1_{2,3}^{f_1}(1, 1, 1; 2; x, y)$ , which gives the error  $E_{2,3}^{f_1}$ . Similarly, displayed in Table 4 are the numerical results for the bivariate function  $f_2(x, y)$  defined by Equation (30). Thus, these results illustrate that the approximation by the truncated Thiele–Newton’s blending expansion is clearly superior in the two test examples.

**Table 3.** Comparison of the numerical results by using  $R_{2,3}^{f_1}(x, y)$  and  $F1_{2,3}^{f_1}(1, 1, 1; 2; x, y)$ .

$(x, y)$	$f_1(x, y)$	$R_{2,3}^{f_1}(x, y)$	$e_{2,3}^{f_1}$	$F1_{2,3}^{f_1}(1, 1, 1; 2; x, y)$	$E_{2,3}^{f_1}$
(0.6,0.5)	2.231435513142	2.175811138576	$5.56244 \times 10^{-2}$	2.007583333333	$2.23852 \times 10^{-1}$
(0.5,0.4)	1.823215567940	1.801574172062	$2.16414 \times 10^{-2}$	1.731400000000	$9.18156 \times 10^{-2}$
(0.4,0.3)	1.541506798273	1.534197264544	$7.30953 \times 10^{-3}$	1.506843333333	$3.46635 \times 10^{-2}$
(0.3,0.2)	1.335313926245	1.333336425463	$1.97750 \times 10^{-3}$	1.324153333333	$1.11606 \times 10^{-2}$
(0.2,0.1)	1.177830356564	1.177455592535	$3.74764 \times 10^{-4}$	1.175210000000	$2.62036 \times 10^{-3}$
(0.09,0.1)	1.104983618659	1.104936257854	$4.73608 \times 10^{-5}$	1.104746383333	$2.37235 \times 10^{-4}$
(0.08,0.09)	1.092907053219	1.092875387558	$3.16657 \times 10^{-5}$	1.092744392933	$1.62660 \times 10^{-4}$
(0.07,0.08)	1.081091610422	1.081071421327	$2.01891 \times 10^{-5}$	1.080985191467	$1.06419 \times 10^{-4}$
(0.05,0.06)	1.058210933054	1.058204252599	$6.68046 \times 10^{-6}$	1.058173883333	$3.70497 \times 10^{-5}$
(0.06,0.05)	1.058210933054	1.058202709844	$8.22321 \times 10^{-6}$	1.058150458333	$6.04747 \times 10^{-5}$
(0.04,0.05)	1.047129986730	1.047126709307	$3.27742 \times 10^{-6}$	1.047111416666	$1.85701 \times 10^{-5}$
(0.05,0.04)	1.047129986730	1.047125552862	$4.43387 \times 10^{-6}$	1.047095800000	$3.41867 \times 10^{-5}$
(0.03,0.02)	1.025650016719	1.025649181797	$8.34922 \times 10^{-7}$	1.025642954533	$7.06219 \times 10^{-6}$
(0.02,0.03)	1.025650016719	1.025649615899	$4.00820 \times 10^{-7}$	1.025647765133	$2.25159 \times 10^{-6}$
(0.02,0.01)	1.015237146402	1.015236912398	$2.34003 \times 10^{-7}$	1.015235095400	$2.05100 \times 10^{-6}$
(0.01,0.02)	1.015237146402	1.015237085235	$6.11671 \times 10^{-8}$	1.015236857467	$2.88935 \times 10^{-7}$

**Table 4.** Comparison of the numerical results by using  $R_{2,3}^{f_2}(x, y)$  and  $F1_{2,3}^{f_2}(1, 2, 2; 2; x, y)$ .

$(x, y)$	$f_2(x, y)$	$R_{2,3}^{f_2}(x, y)$	$e_{2,3}^{f_2}$	$F1_{2,3}^{f_2}(1, 2, 2; 2; x, y)$	$E_{2,3}^{f_2}$
(0.4,0.3)	2.527646365268	2.541958340395	$-1.43120 \times 10^{-2}$	2.314840000000	$2.12806 \times 10^{-1}$
(0.3,0.2)	1.833375742200	1.834880020667	$-1.50428 \times 10^{-3}$	1.774760000000	$5.86157 \times 10^{-2}$
(0.2,0.1)	1.399789684856	1.399902542529	$-1.12858 \times 10^{-4}$	1.387786666667	$1.20030 \times 10^{-2}$
(0.09,0.1)	1.225048763570	1.225046790051	$1.97352 \times 10^{-6}$	1.223971000000	$1.07776 \times 10^{-3}$
(0.08,0.09)	1.197590738753	1.197589594868	$1.14389 \times 10^{-6}$	1.196860955200	$7.29784 \times 10^{-4}$
(0.07,0.08)	1.171129067101	1.171128457172	$6.09930 \times 10^{-7}$	1.170657476267	$4.71591 \times 10^{-4}$
(0.06,0.07)	1.145615802753	1.145615507616	$2.95138 \times 10^{-7}$	1.145329149600	$2.86653 \times 10^{-4}$
(0.05,0.06)	1.121005830888	1.121005702793	$1.28095 \times 10^{-7}$	1.120845560000	$1.60271 \times 10^{-4}$
(0.06,0.05)	1.121005830888	1.121006469601	$-6.38713 \times 10^{-7}$	1.120749100000	$2.56731 \times 10^{-4}$
(0.04,0.05)	1.097256671169	1.097256621117	$5.00525 \times 10^{-8}$	1.097177266667	$7.94045 \times 10^{-5}$
(0.05,0.04)	1.097256671169	1.097256985627	$-3.14458 \times 10^{-7}$	1.097113306667	$1.43365 \times 10^{-4}$
(0.03,0.02)	1.052182967898	1.052183005275	$-3.73770 \times 10^{-8}$	1.052154046400	$2.89215 \times 10^{-5}$
(0.02,0.03)	1.052182967898	1.052182961331	$6.56692 \times 10^{-9}$	1.052173533600	$9.43430 \times 10^{-6}$
(0.02,0.01)	1.030785077555	1.030785083180	$-5.62544 \times 10^{-9}$	1.030776771467	$8.30609 \times 10^{-6}$
(0.01,0.02)	1.030785077555	1.030785075750	$1.80440 \times 10^{-9}$	1.030783868267	$1.20929 \times 10^{-6}$

### 5. Conclusions

From Section 3 in the paper, it is clear to see that we generalized Thiele’s expansion of a univariate function to the bivariate case. Thus, we obtained a rational approximation method, say Thiele–Newton’s blending expansion of a bivariate function. Furthermore, we suggested the Viscovatov-like algorithm, which calculates the coefficients of Thiele–Newton’s expansion and gave the proof of this algorithm. Finally, the application of the Viscovatov-like algorithm was given. Numerical experiments and comparisons were presented in Tables 3 and 4, showing that Thiele–Newton’s blending expansion performed much better approximation than the polynomial expansion. Moreover, the next step in the research work is the consideration of a vector case by a similar technique.

**Author Contributions:** The contributions of all of the authors have been similar. All of them have worked together to develop the present manuscript.

**Funding:** This research was funded by the Project of Leading Talent Introduction and Cultivation in Colleges and Universities of the Education Department of Anhui Province (Grant No. gxfxZD2016270), the Natural Science Key Foundation of the Education Department of Anhui Province (Grant No.KJ2013A183), the Incubation Project of the

National Scientific Research Foundation of Bengbu University (Grant No. 2018GJPY04), and the Project of Quality Curriculums of Education Department of Anhui Province (Grant Nos. 2016gxx087, 2018mooc517).

**Acknowledgments:** The authors are thankful to the anonymous reviewers for their valuable comments.

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

## References

1. Cheney, E.W. *Introduction to Approximation Theory*; McGraw-Hill: New York, NY, USA, 1966.
2. Hildebrand, F.B. *Introduction to Numerical Analysis*, 2nd ed.; McGraw-Hill: New York, NY, USA, 1974.
3. Davis, P.J. *Interpolation and Approximation*; Dover: New York, NY, USA, 1975.
4. Alfio, Q.; Riccardo, S.; Fausto, S. *Numerical Mathematics*; Springer: New York, NY, USA, 2000.
5. Gautschi, W. *Numerical Analysis*, 2nd ed.; Birkhäuser: Boston, FL, USA, 2011.
6. Burden, A.M.; Faires, J.D.; Burden, R.L. *Numerical Analysis*, 10th ed.; Cengage Learning: Boston, FL, USA, 2014.
7. Baker, G.A. *Essentials of Padé Approximants*; Academic Press: New York, NY, USA, 1975.
8. Baker, G.A.; Graves-Morris, P.R. *Padé Approximants*, 2nd ed.; Cambridge University Press: Cambridge, UK, 1996.
9. Kuchminskaya, K. On approximation of functions by continued and branched continued fractions. *Mat. Met. Fiz. Meh. Polya* **1980**, *12*, 3–10.
10. Skorobogatko, V. *Branched Continued Fractions and Their Applications*; Nauka: Moscow, Russian, 1983.
11. Siemaszko, W. Thiele-type branched continued fractions for two variable functions. *J. Comput. Math.* **1983**, *9*, 137–153. [[CrossRef](#)]
12. Viscovatov, B. De la méthode générale pour reduire toutes sortes de quantités en fraction continues. *Mém. Acad. Impériale Sci. St.-Petersbg.* **1805**, *1*, 226–247.
13. Graves-Morris, P.R. Symmetrical formulas for rational interpolants. *J. Comput. Appl. Math.* **1984**, *10*, 107–111. [[CrossRef](#)]
14. Cuyt, A.; Verdonk, B. Multivariate rational interpolants. *Computing* **1985**, *34*, 141–161. [[CrossRef](#)]
15. Cuyt, A.; Verdonk, B. Multivariate reciprocal differences for branched Thiele contiuned fraction expansions. *J. Comput. Appl. Math.* **1988**, *21*, 145–160. [[CrossRef](#)]
16. Cuyt, A.; Verdonk, B. A review of branched contiuned fraction theory for the construction of multivariate rational approximants. *Appl. Numer. Math.* **1988**, *4*, 263–271. [[CrossRef](#)]
17. Cuyt, A.; Driver, K.; Tan, J.Q.; Verdonk, B. Exploring multivariate Padé approximants for multiple hypergeometric series. *Adv. Comput. Math.* **1999**, *10*, 29–49. [[CrossRef](#)]
18. Möller, H.M. Multivariate rational interpolation: Reconstruction of rational functions. *Int. Ser. Numer. Math.* **1989**, *90*, 249–256.
19. Zhu, G.Q.; Tan, J.Q. The duality of vector valued rational interpolants over rectangular grids. *Chin. J. Num. Math. Appl.* **1995**, *17*, 75–84.
20. Gu, C.Q. Bivariate Thiele-type matrix valued rational interpolants. *J. Comput. Appl. Math.* **1997**, *80*, 71–82.
21. Gu, C.Q.; Zhu, G.Q. Bivariate Lagrange-type vector valued rational interpolants. *J. Comput. Math.* **2002**, *2*, 207–216.
22. Tan, J.Q.; Zhu, G.Q. Bivariate vector valued rational interpolants by branched continued fractions. *Numer. Math. A J. Chin. Univ.* **1995**, *4*, 37–43.
23. Tan, J.Q. Bivariate blending rational interpolants. *Approx. Theory Appl.* **1999**, *15*, 74–83.
24. Tan, J.Q. Bivariate rational interpolants with rectangle-hole structure. *J. Comput. Math.* **1999**, *17*, 1–14.
25. Tan, J.Q.; Fang, Y. Newton–Thiele’s rational interpolants. *Numer. Algorithms* **2000**, *24*, 141–157. [[CrossRef](#)]
26. Tan, J.Q. The limiting case of Thiele’s interpolating continued fraction expansion. *J. Comput. Math.* **2001**, *19*, 433–444.
27. Tan, J.Q.; Jiang, P. A Neville-like method via continued fractions. *J. Comput. Appl. Math.* **2004**, *163*, 219–232. [[CrossRef](#)]
28. Tan, J.Q. *The Theory of Continued Fractions and Their Applications*; Science Press: Beijing, China, 2007.
29. Hu, M.; Tan, J.Q.; Xue, F. A new approach to the image resizing using interpolating rational-linear splines by continued fractions. *J. Inf. Comput. Sci.* **2005**, *2*, 681–685.

30. Hu, M.; Tan, J.Q. Adaptive osculatory rational interpolation for image processing. *J. Comput. Appl. Math.* **2006**, *195*, 46–53. [[CrossRef](#)]
31. Li, S.F.; Song, L.T.; Xie, J.; Dong, Y. Image inpainting based on bivariate interpolation by continued fractions. In Proceedings of the IEEE International Conference on Computer Science and Automation Engineering, Zhangjiajie, China, 25–27 May 2012; Volume 2, pp. 756–759.
32. Li, S.F.; Dong, Y.  $k$ -Hypergeometric series solutions to one type of non-homogeneous  $k$ -hypergeometric equations. *Symmetry* **2019**, *11*, 262. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).