MATHEMATICS OF COMPUTATION
Volume 69, Number 232, Pages 1521–1532
S 0025-5718(99)01170-9
Article electronically published on August 24, 1999

# BIVARIATE COMPOSITE VECTOR VALUED RATIONAL INTERPOLATION

JIEQING TAN AND SHUO TANG

ABSTRACT. In this paper we point out that bivariate vector valued rational interpolants (BVRI) have much to do with the vector-grid to be interpolated. When a vector-grid is well-defined, one can directly design an algorithm to compute the BVRI. However, the algorithm no longer works if a vector-grid is ill-defined. Taking the policy of "divide and conquer", we define a kind of bivariate composite vector valued rational interpolant and establish the corresponding algorithm. A numerical example shows our algorithm still works even if a vector-grid is ill-defined.

## 1. MOTIVATION

Let $\{(x_i, y_j)|i, j = 0, 1, \ldots, n\}$ be a set of points in $\mathbb{R}^2$ and let these points be reordered into the following array

$$\begin{matrix} (x_0, y_0) & \cdots & (x_0, y_n) \\ \vdots & \ddots & \vdots \\ (x_n, y_0) & \cdots & (x_n, y_n) \end{matrix} \tag{1.1}$$

where $x_i > x_{i+1}$ and $y_i < y_{i+1}$ for $i = 0, 1, \cdots, n-1$. We call this array a *square point-grid*, and denote it by $\Pi^n$. Let $\vec{v}_{i,j} \in \mathbb{C}^d$ be a $d$-dimensional vector associated with the point $(x_i, y_j) \in \Pi^n$. Similarly we arrange these $\vec{v}_{i,j}$ into the following array

$$\begin{matrix} \vec{v}_{0,0} & \vec{v}_{0,1} & \cdots & \vec{v}_{0,n} \\ \vec{v}_{1,0} & \vec{v}_{1,1} & \cdots & \vec{v}_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{v}_{n,0} & \vec{v}_{n,1} & \cdots & \vec{v}_{n,n} \end{matrix} \tag{1.2}$$

and call it a vector-grid, denoted by $\vec{V}^n$.

For a $d$-dimensional vector $\vec{v} = (v_1, v_2, \cdots, v_d) \in \mathbb{C}^d$, its generalized inverse (or the Samelson inverse) is defined as (see [3])

$$\vec{v}^{-1} = \frac{(v_1^*, v_2^*, \cdots, v_d^*)}{\sum_{i=1}^d v_i v_i^*}, \tag{1.3}$$

where $v_i^*$ denotes the complex conjugate of $v_i$.

Received by the editor April 10, 1997 and, in revised form, December 10, 1998.
1991 *Mathematics Subject Classification.* Primary 41A20; Secondary 65D05.
*Key words and phrases.* Branched continued fraction, interpolation, algorithm.
Supported by the National Natural Science Foundation of China and in part by the Spanning-the-Century Foundation for Excellent Talents of the Ministry of the Machine Building Industry of China.

**Definition 1.1.** A $d$-dimensional vector valued polynomial

$$\vec{N}(x,y) = (N_1(x,y), N_2(x,y), \cdots, N_d(x,y))$$

is said to be *of degree* $n$ and we write $\partial\vec{N} = n$, if $\partial N_i(x,y) \le n$ for $i = 1,2,\cdots,d$ and $\partial N_j(x,y) = n$ for some $j$ $(1 \le j \le d)$.

**Definition 1.2.** Denote by $H_n$ the collection of all bivariate polynomials with total degree not exceeding $n$ and by $\vec{H}_n$ the collection of $d$ dimensional bivariate vector valued polynomials of degree $n$. Then the set

$$\vec{H}_{n,m} = \{\vec{N}(x,y)/M(x,y) | \vec{N}(x,y) \in \vec{H}_n, M(x,y) \in H_m\}$$

is called *the collection of bivariate vector valued rational functions of type* $(n/m)$.

All vectors in this paper are $d$-dimensional unless otherwise specified.

Making use of the Samelson inverse and reciprocal difference, one of the authors constructed the following Thiele-type branched continued fraction (see [5]):

$$\vec{r}_n(x,y) = \vec{t}_0(y) + \frac{x-x_0}{\vec{t}_1(y)} + \cdots + \frac{x-x_{n-1}}{\vec{t}_n(y)}, \tag{1.4}$$

where

$$\begin{aligned}\vec{t}_l(y) &= \vec{c}_{l,0}(\overline{x_0,\cdots,x_l};y_0)\\ &+ \frac{y-y_0}{\vec{c}_{l,1}(\overline{x_0,\cdots,x_l};y_0,y_1)} + \cdots + \frac{y-y_{n-1}}{\vec{c}_{l,n}(\overline{x_0,\cdots,x_l};y_0,\cdots,y_n)},\end{aligned} \tag{1.5}$$

and $\vec{c}_{i,j}(\overline{x_0,\cdots,x_i};y_0,\cdots,y_j)$ are computed through the following recursive process

$$\vec{c}_{0,0}(\bar{x}_i,y_j) = \vec{v}_{i,j} \qquad (i = 0,1,\cdots,n,\ j = 0,1,\cdots,n), \tag{1.6}$$

$$\vec{c}_{0,j}(\bar{x}_i;y_0,\cdots,y_j) = \frac{y_j - y_{j-1}}{\vec{c}_{0,j-1}(\bar{x}_i;y_0,\cdots,y_{j-2},y_j) - \vec{c}_{0,j-1}(\bar{x}_i;y_0,\cdots,y_{j-2},y_{j-1})}, \tag{1.7}$$

$$\vec{c}_{i,0}(\overline{x_0,\cdots,x_i};y_j) = \frac{x_i - x_{i-1}}{\vec{c}_{i-1,0}(\overline{x_0,\cdots,x_{i-2},x_i};y_j) - \vec{c}_{i-1,0}(\overline{x_0,\cdots,x_{i-2},x_{i-1}};y_j)}, \tag{1.8}$$

$$\begin{aligned}&\vec{c}_{i,j}(\overline{x_0,\cdots,x_i};y_0,\cdots,y_j)\\ &= \frac{y_j - y_{j-1}}{\vec{c}_{i,j-1}(\overline{x_0,\cdots,x_i};y_0,\cdots,y_{j-2},y_j) - \vec{c}_{i,j-1}(\overline{x_0,\cdots,x_i};y_0,\cdots,y_{j-2},y_{j-1})}.\end{aligned} \tag{1.9}$$

It is not difficult to prove (see [5]) that $\vec{r}_n(x,y) \in \vec{H}_{n^2+2n,2[(n^2+2n)/2]}$ (here $[x]$ denotes the greatest integer not exceeding $x$) and

$$\vec{r}_n(x_i,y_j) = \vec{v}_{i,j} \qquad (i = 0,1,\cdots,n,\ j = 0,1,\cdots,n). \tag{1.10}$$

If the roles of $x$ and $y$ are interchanged, one will obtain a so-called dual Thiele-type branched continued fraction (see [5])

$$\vec{r}_n^{*}(x,y) = \vec{t}_0^{*}(x) + \frac{y-y_0}{\vec{t}_1^{*}(x)} + \cdots + \frac{y-y_{n-1}}{\vec{t}_n^{*}(x)}, \tag{1.11}$$

where

$$\vec{t}_l^*(x) = \vec{c}_{0,l}^*(x_0; \overline{y_0, \cdots, y_l}) + \frac{x - x_0}{\vec{c}_{1,l}^*(x_0, x_1; \overline{y_0, \cdots, y_l})} + \cdots + \frac{x - x_{n-1}}{\vec{c}_{n,l}^*(x_0, \cdots, x_n; \overline{y_0, \cdots, y_l})} \tag{1.12}$$

and $\vec{c}_{i,j}^*(x_0, \cdots, x_i; \overline{y_0, \cdots, y_j})$ are computed according to the following recursive process

$$\vec{c}_{0,0}^*(x_i, \bar{y}_j) = \vec{v}_{i,j} \qquad (i = 0, 1, \cdots, n,\ j = 0, 1, \cdots, n), \tag{1.13}$$

(1.14)

$$\vec{c}_{0,j}^*(x_i; \overline{y_0, \cdots, y_j}) = \frac{y_j - y_{j-1}}{\vec{c}_{0,j-1}^*(x_i; \overline{y_0, \cdots, y_{j-2}, y_j}) - \vec{c}_{0,j-1}^*(x_i; \overline{y_0, \cdots, y_{j-2}, y_{j-1}})},$$

(1.15)

$$\vec{c}_{i,0}^*(x_0, \cdots, x_i; \bar{y}_j) = \frac{x_i - x_{i-1}}{\vec{c}_{i-1,0}^*(x_0, \cdots, x_{i-2}, x_i; \bar{y}_j) - \vec{c}_{i-1,0}^*(x_0, \cdots, x_{i-2}, x_{i-1}; \bar{y}_j)},$$

(1.16)

$$\vec{c}_{i,j}^*(x_0, \cdots, x_i; \overline{y_0, \cdots, y_j}) = \frac{x_i - x_{i-1}}{\vec{c}_{i-1,j}^*(x_0, \cdots, x_{i-2}, x_i; \overline{y_0, \cdots, y_j}) - \vec{c}_{i-1,j}(x_0, \cdots, x_{i-2}, x_{i-1}; \overline{y_0, \cdots, y_j})}.$$

To distinguish $\vec{r}_n(x, y)$ from $\vec{r}_n^*(x, y)$, we might as well call $\vec{r}_n(x, y)$ defined in (1.4)–(1.9) an *$x/y$-type* and $\vec{r}_n^*(x, y)$ defined in (1.11)–(1.16) a *$y/x$-type.* It can be proved that $\vec{r}_n^*(x, y) \in \vec{H}_{n^2+2n, 2[(n^2+2n)/2]}$ and

$$\vec{r}_n^*(x_i, y_j) = \vec{v}_{i,j} \qquad (i = 0, 1, \ldots, n,\ j = 0, 1, \ldots, n). \tag{1.17}$$

Although both $\vec{r}_n(x, y)$ and $\vec{r}_n^*(x, y)$ are of the same rational type and have the same interpolation properties, one can by no means assert that $\vec{r}_n(x, y) \equiv \vec{r}_n^*(x, y)$, as is shown by a numerical example in [5]. However, if the square point-grid $\Pi^n$ is symmetric, by which we mean $x_i = y_i$ for $i = 0, 1, \ldots, n$, and the vector-grid $\vec{V}^n$ is symmetric, by which we mean $\vec{v}_{i,j} = \vec{v}_{j,i}$ for $i, j = 0, 1, \ldots, n$, then we can conclude $\vec{r}_n(x, y) \equiv \vec{r}_n^*(y, x)$ (see [5]).

In what follows, we only restrict ourselves to the discussion of $x/y$-type bivariate vector valued rational interpolants (BVRI), because the results in $x/y$-type BVRI can easily be transplanted into $y/x$-type.

For convenience, let us simply set

$$\vec{c}_{i,j}^{(i,j)} = \vec{c}_{i,j}(\overline{x_0, \ldots, x_i}; y_0, \ldots, y_j) \qquad (i = 0, 1, \ldots, n,\ j = 0, 1, \ldots, n). \tag{1.18}$$

Then we have the following algorithm to compute $\vec{r}_n(x, y)$.

**Algorithm 1.1.** *This algorithm is carried out according to the following three steps.*

a) *For $i = 0, 1, \cdots, n$ and $j = 0, 1, \cdots, n$, let*

$$\vec{c}_{i,j}^{(0,0)} = \vec{v}_{i,j}.$$

b) *For* $j = 0, 1, \cdots, n,\ p = 1, 2, \cdots, n,$ *and* $i = p, p+1, \cdots, n,$ *let*

$$\vec{c}_{i,j}^{(p,0)} = \frac{x_i - x_{p-1}}{\vec{c}_{i,j}^{(p-1,0)} - \vec{c}_{p-1,j}^{(p-1,0)}}.$$

c) *For* $i = 0, 1, \cdots, n,\ q = 1, 2, \cdots, n,$ *and* $j = q, q+1, \cdots, n,$ *let*

$$\vec{c}_{i,j}^{(i,q)} = \frac{y_j - y_{q-1}}{\vec{c}_{i,j}^{(i,q-1)} - \vec{c}_{i,q-1}^{(i,q-1)}}.$$

It is easy to verify that

$$\vec{r}_n(x_i, y_j) = \vec{v}_{i,j}, \quad \forall (x_i, y_j) \in \Pi^n.$$

**Definition 1.3.** A vector-grid $\vec{V}^n$ is said to be *well-defined* if the $\vec{c}_{i,j}^{(p,q)}$ as defined in Algorithm 1.1 satisfy $\vec{c}_{i,j}^{(p-1,0)} \neq \vec{c}_{p-1,j}^{(p-1,0)}$ for $j = 0, 1, \ldots, n,\ p = 1, 2, \ldots, n,$ and $i = p, p+1, \ldots, n,$ and $\vec{c}_{i,j}^{(i,q-1)} \neq \vec{c}_{i,q-1}^{(i,q-1)}$ for $i = 0, 1, \ldots, n,\ q = 1, 2, \ldots, n,$ and $j = q, q+1, \ldots, n$. Otherwise the grid $\vec{V}^n$ is said to be *ill-defined.*

It is clear that if a vector-grid $\vec{V}^n$ is ill-defined, then Algorithm 1.1 does not work any more.

**Example 1.1.** Let $\Pi^1$ and a two-dimensional vector-grid $\vec{V}^1$ be given as follows:

$$\Pi^1 : \begin{array}{cc} (1,0) & (1,1) \\ (0,0) & (0,1), \end{array}$$

$$\vec{V}^1 : \begin{array}{cc} (1,0) & (1,1) \\ (0,0) & (1,0). \end{array}$$

Proceeding by Algorithm 1.1, we obtain

$$\begin{array}{cc} \vec{c}_{0,0}^{(0,0)} = (1,0) & \vec{c}_{0,1}^{(0,0)} = (1,1) \\ \vec{c}_{1,0}^{(0,0)} = (0,0) & \vec{c}_{1,1}^{(0,0)} = (1,0) \\ & \downarrow \\ \vec{c}_{0,0}^{(0,0)} = (1,0) & \vec{c}_{0,1}^{(0,0)} = (1,1) \\ \vec{c}_{1,0}^{(1,0)} = (1,0) & \vec{c}_{1,1}^{(1,0)} = (0,1) \\ & \downarrow \\ \vec{c}_{0,0}^{(0,0)} = (1,0) & \vec{c}_{0,1}^{(0,1)} = (0,1) \\ \vec{c}_{1,0}^{(1,0)} = (1,0) & \vec{c}_{1,1}^{(1,1)} = (-\frac{1}{2}, \frac{1}{2}). \end{array}$$

Obviously $\vec{V}^1$ is well-defined. As a result,

$$\begin{aligned} \vec{t}_0(y) &= \vec{c}_{0,0} + \frac{y - y_0}{\vec{c}_{0,1}} = (1,0) + \frac{y}{(0,1)} = (1, y), \\ \vec{t}_1(y) &= \vec{c}_{1,0} + \frac{y - y_0}{\vec{c}_{1,1}} = (1,0) + \frac{y}{(-\frac{1}{2}, \frac{1}{2})} = (1 - y, y). \end{aligned}$$

Consequently we get

$$\begin{aligned} \vec{r}_1(x, y) &= \vec{t}_0(y) + \frac{x - x_0}{\vec{t}_1(y)} = (1, y) + \frac{x - 1}{(1 - y, y)} \\ &= \frac{(y^2 + (1-y)^2 + (x-1)(1-y), y(y^2 + (1-y)^2 + x - 1))}{(1-y)^2 + y^2}. \end{aligned}$$

**Example 1.2.** Let $\Pi^2$ and the two-dimensional vector-grid $\vec{V}^2$ be given as follows:

$$\Pi^2 : \quad \begin{array}{ccc} (0,0) & (0,1) & (0,2) \\ (-1,0) & (-1,1) & (-1,2) \\ (-2,0) & (-2,1) & (-2,2), \end{array}$$

$$\vec{V}^2 : \quad \begin{array}{ccc} (2,2) & (6,0) & (24,24) \\ (12,6) & (6,0) & (12,6) \\ (0,0) & (6,0) & (-2,2). \end{array}$$

We see $\vec{v}_{0,1} = \vec{v}_{1,1} = \vec{v}_{2,1} = (6,0)$, which leads to $\vec{c}_{0,1}^{(0,0)} = \vec{c}_{1,1}^{(0,0)} = \vec{c}_{2,1}^{(0,0)}$; therefore $\vec{V}^2$ is ill-defined and we cannot use Algorithm 1.1 to construct a vector-valued rational function $\vec{r}_2(x,y)$ that interpolates $\vec{V}^2$ over $\Pi^2$.

In the next section, we define a new interpolant with a corresponding algorithm more reliable than Algorithm 1.1.

## 2. The definition and computation of BCVRI

Let us decompose the grid $\Pi^n$ into the following two triangular grids:

$$\begin{array}{cccc} (x_0,y_0) & & & \\ (x_1,y_0) & (x_1,y_1) & & \\ \vdots & \vdots & \ddots & \\ (x_n,y_0) & (x_n,y_1) & \cdots & (x_n,y_n) \end{array} \tag{2.1}$$

and

$$\begin{array}{cccc} (x_0,y_1) & (x_0,y_2) & \cdots & (x_0,y_n) \\ & (x_1,y_2) & \cdots & (x_1,y_n) \\ & & \ddots & \vdots \\ & & & (x_{n-1},y_n), \end{array} \tag{2.2}$$

denoted by LB and RU, respectively. We hope to use the policy of "divide and conquer" to construct a kind of composite vector valued rational interpolant. In what follows we abbreviate the term *bivariate composite vector valued rational interpolant* as BCVRI.

Let

$$\vec{R}_n(LB;x,y) = \vec{S}_0(LB;y) + \frac{x-x_n \,|}{|\, \vec{S}_1(LB;y)} + \cdots + \frac{x-x_1 \,|}{|\, \vec{S}_n(LB;y)}, \tag{2.3}$$

$$\vec{R}_n(RU;x,y) = \vec{S}_0(RU;y) + \frac{x-x_0 \,|}{|\, \vec{S}_1(RU;y)} + \cdots + \frac{x-x_{n-2} \,|}{|\, \vec{S}_{n-1}(RU;y)}, \tag{2.4}$$

where

$$\vec{S}_k(LB;y) = \vec{a}_{k,0} + \frac{y-y_0 \,|}{|\, \vec{a}_{k,1}} + \cdots + \frac{y-y_{n-k-1} \,|}{|\, \vec{a}_{k,n-k}}, \qquad k = 0,1,\cdots,n, \tag{2.5}$$

(2.6)

$$\vec{S}_k(RU;y) = \vec{b}_{k,k+1} + \frac{y-y_{k+1} \,|}{|\, \vec{b}_{k,k+2}} + \cdots + \frac{y-y_{n-1} \,|}{|\, \vec{b}_{k,n}}, \qquad k = 0,1,\cdots,n-1.$$

Suppose $\Pi^n$ is uniform, i.e.,

$$x_{i-1} - x_i = x_i - x_{i+1} = y_{i+1} - y_i = y_i - y_{i-1}, \qquad i = 1, 2, \cdots, n-1, \tag{2.7}$$

and let

$$P(x, y) = \prod_{i=0}^{n} (x + y - x_n - y_i), \tag{2.8}$$

$$Q(x, y) = \prod_{i=0}^{n-1} (x + y - x_i - y_n). \tag{2.9}$$

It is clear that $P(x, y)$ and $Q(x, y)$ are polynomials of degree $n + 1$ and $n$, respectively, and

$$\begin{array}{lll} P(x_i, y_j) = 0, & Q(x_i, y_j) \neq 0 & \text{if } (x_i, y_j) \in \text{LB}, \\ P(x_i, y_j) \neq 0, & Q(x_i, y_j) = 0 & \text{if } (x_i, y_j) \in \text{RU}. \end{array} \tag{2.10}$$

When $\Pi^n$ is not uniform, by which we mean that the conditions (2.7) are not satisfied, one can also construct polynomials $P(x, y)$ and $Q(x, y)$ such that (2.10) holds. In general, however, the degrees of the polynomials will be much higher. For example,

$$\begin{aligned} P(x, y) &= \prod_{(x_i, y_j) \in \text{LB}} [(x - x_i)^2 + (y - y_j)^2] \\ Q(x, y) &= \prod_{(x_i, y_j) \in \text{RU}} [(x - x_i)^2 + (y - y_j)^2] \end{aligned}$$

are the polynomials satisfying (2.10) with degree $(n + 1)(n + 2)$ and $n(n + 1)$, respectively.

Now we define a BCVRI over $\Pi^n$ as follows:

$$\vec{R}_n(x, y) = Q(x, y)\vec{R}_n(\text{LB}; x, y) + P(x, y)\vec{R}_n(\text{RU}; x, y). \tag{2.11}$$

The following algorithm aims at computing the coefficients $\vec{a}_{k,l}$ and $\vec{b}_{k,l}$ in branched continued fractions $\vec{R}_n(\text{LB}; x, y)$ and $\vec{R}_n(\text{RU}; x, y)$ simultaneously.

**Algorithm 2.1.** *This algorithm proceeds as follows.*

a) *For $i = 0, 1, \cdots, n$ and $j = 0, 1, \cdots, i$, let*

$$\vec{A}_{i,j}^{(0,0)} = \vec{v}_{i,j} / Q(x_i, y_j). \tag{2.12}$$

*For $i = 0, 1, \cdots, n-1$ and $j = i+1, i+2, \cdots, n$, let*

$$\vec{B}_{i,j}^{(0,0)} = \vec{v}_{i,j} / P(x_i, y_j). \tag{2.13}$$

b) *For $j = 0, 1, \cdots, n$, $p = 1, 2, \cdots, n-j$, and $i = j, j+1, \cdots, n-p$, let*

$$\vec{A}_{i,j}^{(p,0)} = \frac{x_i - x_{n-p+1}}{\vec{A}_{i,j}^{(p-1,0)} - \vec{A}_{n-p+1,j}^{(p-1,0)}}. \tag{2.14}$$

c) *For $i = 0, 1, \cdots, n$, $q = 1, 2, \cdots, i$, and $j = q, q+1, \cdots, i$, let*

$$\vec{A}_{i,j}^{(n-i,q)} = \frac{y_j - y_{q-1}}{\vec{A}_{i,j}^{(n-i,q-1)} - \vec{A}_{i,q-1}^{(n-i,q-1)}}. \tag{2.15}$$

d) *For* $j = 1, 2, \cdots, n,\ p = 1, 2, \cdots, j-1,$ *and* $i = p, p+1, \cdots, j-1,$ *let*

$$\vec{B}_{i,j}^{(p,0)} = \frac{x_i - x_{p-1}}{\vec{B}_{i,j}^{(p-1,0)} - \vec{B}_{p-1,j}^{(p-1,0)}}. \tag{2.16}$$

e) *For* $i = 0, 1, \cdots, n-1$ *and* $j = i+1, i+2, \cdots, n,$ *let*

$$\vec{B}_{i,j}^{(i,i+1)} = \vec{B}_{i,j}^{(i,0)}. \tag{2.17}$$

f) *For* $i = 0, 1, \cdots, n-2,\ j = i+2, i+3, \cdots, n,$ *and* $q = i+2, i+3, \cdots, j,$ *let*

$$\vec{B}_{i,j}^{(i,q)} = \frac{y_j - y_{q-1}}{\vec{B}_{i,j}^{(i,q-1)} - \vec{B}_{i,q-1}^{(i,q-1)}}. \tag{2.18}$$

**Theorem 2.1.** *Let*

$$\vec{a}_{k,l} = \vec{A}_{n-k,l}^{(k,l)} \qquad (k = 0, 1, \cdots, n,\ l = 0, 1, \cdots, n-k), \tag{2.19}$$

$$\vec{b}_{k,l} = \vec{B}_{k,l}^{(k,l)} \qquad (k = 0, 1, \cdots, n-1, l = k+1, \cdots, n). \tag{2.20}$$

*Then*

$$\vec{R}_n(x,y) \in \vec{H}_{\max(\partial Q + n^2+n-2, \partial P + n^2+n-1), n^2+n-2} \quad \textit{for even } n,$$
$$\vec{R}_n(x,y) \in \vec{H}_{\max(\partial Q + n^2+n-1, \partial P + n^2+n-2), n^2+n-2} \quad \textit{for odd } n,$$

*where* $\partial Q$ *and* $\partial P$ *denote the total degrees of polynomials* $Q(x,y)$ *and* $P(x,y)$, *respectively. (In particular, if* $\Pi^n$ *is uniform, then* $\partial Q = n$ *and* $\partial P = n+1$. *In this case,* $\vec{R}_n(x,y) \in \vec{H}_{n^2+2n, n^2+n-2}$ *for even* $n$ *and* $\vec{R}_n(x,y) \in \vec{H}_{n^2+2n-1, n^2+n-2}$ *for odd* $n$.*) Moreover,*

$$\vec{R}_n(x_i, y_j) = \vec{v}_{i,j} \qquad \forall (x_i, y_j) \in \Pi^n.$$

*Proof.* It is not difficult to show by induction that

$$\vec{R}_n(LB; x, y) \in \vec{H}_{(n^2+2n)/2, (n^2+2n)/2} \quad \text{for even } n,$$
$$\vec{R}_n(LB; x, y) \in \vec{H}_{(n^2+2n-1)/2, (n^2+2n-3)/2} \quad \text{for odd } n,$$
$$\vec{R}_n(RU; x, y) \in \vec{H}_{(n^2-2)/2, (n^2-4)/2} \quad \text{for even } n,$$
$$\vec{R}_n(RU; x, y) \in \vec{H}_{(n^2-1)/2, (n^2-1)/2} \quad \text{for odd } n.$$

Therefore

$$\vec{R}_n(x,y) \in \vec{H}_{\max(\partial Q + n^2+n-2, \partial P + n^2+n-1), n^2+n-2} \quad \text{for even } n$$
$$\vec{R}_n(x,y) \in \vec{H}_{\max(\partial Q + n^2+n-1, \partial P + n^2+n-2), n^2+n-2} \quad \text{for odd } n.$$

Since $\Pi^n = \text{LB} \cup \text{RU}$, $(x_i, y_j) \in \Pi^n$ implies $(x_i, y_j) \in \text{LB}$ or $(x_i, y_j) \in \text{RU}$. If $(x_i, y_j) \in \text{LB}$, then from (2.5), (2.19) and (2.15) it follows that

$$\begin{aligned}
\vec{S}_k(\text{LB}; y_j) &= \vec{A}_{n-k,0}^{(k,0)} + \frac{y_j - y_0|}{|\vec{A}_{n-k,1}^{(k,1)}} + \cdots + \frac{y_j - y_{j-1}|}{|\vec{A}_{n-k,j}^{(k,j)}} \\
&= \vec{A}_{n-k,0}^{(k,0)} + \frac{y_j - y_0|}{|\vec{A}_{n-k,1}^{(k,1)}} + \cdots + \frac{y_j - y_{j-2}|}{|\vec{A}_{n-k,j}^{(k,j-1)}} \\
&= \cdots = \vec{A}_{n-k,j}^{(k,0)}.
\end{aligned}$$

By (2.3), (2.14) and (2.12) one has

$$\begin{aligned}
\vec{R}_n(\mathrm{LB}; x_i, y_j) &= \vec{A}^{(0,0)}_{n,j} + \frac{x_i - x_n |}{| \vec{A}^{(1,0)}_{n-1,j}} + \cdots + \frac{x_i - x_{i+1} |}{| \vec{A}^{(n-i,0)}_{i,j}} \\
&= \vec{A}^{(0,0)}_{n,j} + \frac{x_i - x_n |}{| \vec{A}^{(1,0)}_{n-1,j}} + \cdots + \frac{x_i - x_{i+2} |}{| \vec{A}^{(n-i-1,0)}_{i,j}} \\
&= \cdots = \vec{A}^{(0,0)}_{i,j} = \vec{v}_{i,j}/Q(x_i, y_j).
\end{aligned}$$

Therefore, by (2.11) one finally gets

$$\vec{R}_n(x_i, y_j) = Q(x_i, y_j)\vec{R}_n(\mathrm{LB}; x_i, y_j) = \vec{v}_{i,j}.$$

If $(x_i, y_j) \in \mathrm{RU}$, then from (2.6), (2.20) and (2.18) it follows that

$$\begin{aligned}
\vec{S}_k(\mathrm{RU}; y_j) &= \vec{B}^{(k,k+1)}_{k,k+1} + \frac{y_j - y_{k+1} |}{| \vec{B}^{(k,k+2)}_{k,k+2}} + \cdots + \frac{y_j - y_{j-1} |}{| \vec{B}^{(k,j)}_{k,j}} \\
&= \vec{B}^{(k,k+1)}_{k,k+1} + \frac{y_j - y_{k+1} |}{| \vec{B}^{(k,k+2)}_{k,k+2}} + \cdots + \frac{y_j - y_{j-2} |}{| \vec{B}^{(k,j-1)}_{k,j}} \\
&= \cdots = \vec{B}^{(k,k+1)}_{k,j}.
\end{aligned}$$

Thus, from (2.4), (2.17), (2.16) and (2.13) we get

$$\begin{aligned}
\vec{R}_n(\mathrm{RU}; x_i, y_j) &= \vec{B}^{(0,1)}_{0,j} + \frac{x_i - x_0 |}{| \vec{B}^{(1,2)}_{1,j}} + \cdots + \frac{x_i - x_{i-1} |}{| \vec{B}^{(i,i+1)}_{i,j}} \\
&= \vec{B}^{(0,0)}_{0,j} + \frac{x_i - x_0 |}{| \vec{B}^{(1,0)}_{1,j}} + \cdots + \frac{x_i - x_{i-1} |}{| \vec{B}^{(i,0)}_{i,j}} \\
&= \vec{B}^{(0,0)}_{0,j} + \frac{x_i - x_0 |}{| \vec{B}^{(1,0)}_{1,j}} + \cdots + \frac{x_i - x_{i-2} |}{| \vec{B}^{(i-1,0)}_{i,j}} \\
&= \cdots = \vec{B}^{(0,0)}_{i,j} = \vec{v}_{i,j}/P(x_i, y_j).
\end{aligned}$$

Hence, by (2.11) we have

$$\vec{R}_n(x_i, y_j) = P(x_i, y_j)\vec{R}_n(\mathrm{RU}; x_i, y_j) = \vec{v}_{i,j}.$$

The proof is completed. □

## 3. The complexity of algorithms

Instead of (2.1) and (2.2) one can also carry out other triangular decompositions of the square grid, for instance, the decomposition

$$\begin{array}{llll}
(x_1, y_0) & & & \\
(x_2, y_0) & (x_2, y_1) & & \\
\vdots & \vdots & \ddots & \\
(x_n, y_0) & (x_n, y_1) & \cdots & (x_n, y_{n-1})
\end{array}
\qquad
\begin{array}{llll}
(x_0, y_0) & (x_0, y_1) & \cdots & (x_0, y_n) \\
 & (x_1, y_1) & \cdots & (x_1, y_n) \\
 & & \ddots & \vdots \\
 & & & (x_n, y_n)
\end{array}$$

and the decomposition, along another diagonal,

$$\begin{array}{cccccccc} (x_0,y_0) & \cdots & (x_0,y_{n-1}) & (x_0,y_n) & & & & (x_1,y_n) \\ (x_1,y_0) & \cdots & (x_1,y_{n-1}) & & & & (x_2,y_{n-1}) & (x_2,y_n) \\ \vdots & & & & & & \vdots & \vdots \\ (x_n,y_0) & & & & (x_n,y_1) & \cdots & (x_n,y_{n-1}) & (x_n,y_n). \end{array}$$

It is not difficult to define the corresponding BCVRIs based on the above decompositions which interpolate $\vec{V}^n$ over $\Pi^n$.

For a vector valued continued fraction, the complexity is obviously related to the computation of the Samelson inverses. From (1.3) we know that carrying out a Samelson inversion for a $d$-dimensional vector demands at least $2d$ operations of multiplications or divisions. Therefore we take the number of Samelson inverses in an algorithm as the criterion for judging whether the algorithm is complicated or not.

Suppose $N_1$ and $N_2$ are the total numbers of Samelson inverses to be computed for the vector valued rational interpolants of form (2.11) and (1.4)–(1.5), respectively. Then

$$\begin{aligned} N_1 &= \sum_{j=0}^{n} \frac{(n-j)(n-j+1)}{2} + \sum_{i=0}^{n} \frac{i(i+1)}{2} + \sum_{j=1}^{n} \frac{j(j-1)}{2} + \sum_{i=0}^{n-2} \frac{(n-i+1)(n-i)}{2} \\ &= \frac{n(n+1)(n+2)}{6} + \frac{n(n+1)(n+2)}{6} + \frac{(n-1)n(n+1)}{6} + \frac{(n-1)n(n+1)}{6} \\ &= \frac{n(n+1)(2n+1)}{3}, \end{aligned}$$

$$N_2 = n(n+1)^2,$$

which shows that it is $n(n+1)(n+2)/3$ times more economical to compute the BCVRI $\vec{R}_n(x,y)$ in (2.11) yielded by the decomposition (2.1) and (2.2) of $\Pi^n$ than to compute (1.4) and (1.5) directly. Therefore at least $2n(n+1)(n+2)d/3$ multiplications are saved through our decomposition method.

## 4. Numerical example

Let us consider again the grid $\Pi^2$ and the corresponding vector-grid $\vec{V}^2$ in Example 1.2, i.e.,

$$\Pi^2: \quad \begin{array}{ccc} (0,0) & (0,1) & (0,2) \\ (-1,0) & (-1,1) & (-1,2) \\ (-2,0) & (-2,1) & (-2,2), \end{array}$$

$$\vec{V}^2: \quad \begin{array}{ccc} (2,2) & (6,0) & (24,24) \\ (12,6) & (6,0) & (12,6) \\ (0,0) & (6,0) & (-2,2). \end{array}$$

In this case, $\Pi^2$ is uniform and $\vec{V}^2$ is ill-defined. We mentioned in Example 1.2 that the computational procedure in Algorithm 1.1 breaks down. In fact, $\vec{r}_2(x,y)$ does not exist at all in this case. Otherwise, $\vec{r}_2(x,y)$ can be written as

$$\vec{r}_2(x,y) = \vec{t}_0(y) + \frac{x-x_0}{\vec{t}_1(y)} \; + \; \frac{x-x_1}{\vec{t}_2(y)}.$$

Whatever a reordering of the square point-grid $\Pi^2$ and vector-grid $\vec{V}^2$ is made, we always have a whole column in $\vec{V}^2$, entries of which are all equal to $(6,0)$, i.e., $\vec{v}_{0,j} = \vec{v}_{1,j} = \vec{v}_{2,j} = (6,0)$ with some $j$ in $\{0,1,2\}$. Therefore we have

$$\vec{r}_2(x_0, y_j) = \vec{r}_2(x_1, y_j) = \vec{r}_2(x_2, y_j),$$

which leads to

$$\vec{t}_0(y_j) + \frac{x_1 - x_0}{\vec{t}_1(y_j)} = \vec{t}_0(y_j).$$

The above relations imply $(x_1 - x_0)/\vec{t}_1(y_j) = 0$ which is impossible because $x_1 \neq x_0$.

Next, we turn to the construction of a BCVRI defined in (2.11). By (2.8) and (2.9),

$$\begin{aligned} P(x,y) &= (x+y+2)(x+y+1)(x+y), \\ Q(x,y) &= (x+y-2)(x+y-1). \end{aligned}$$

By (2.12) and (2.13), one gets

$$\begin{aligned} &\vec{A}^{(0,0)}_{0,0} = (1,1), \\ &\vec{A}^{(0,0)}_{1,0} = (2,1), \quad \vec{A}^{(0,0)}_{1,1} = (3,0), \\ &\vec{A}^{(0,0)}_{2,0} = (0,0), \quad \vec{A}^{(0,0)}_{2,1} = (1,0), \quad \vec{A}^{(0,0)}_{2,2} = (-1,1) \end{aligned}$$

and

$$\begin{aligned} \vec{B}^{(0,0)}_{0,1} = (1,0), \quad &\vec{B}^{(0,0)}_{0,2} = (1,1), \\ &\vec{B}^{(0,0)}_{1,2} = (2,1). \end{aligned}$$

According to (2.14) and (2.15), one obtains in order

$$\begin{aligned} &\vec{A}^{(1,0)}_{0,0} = (1,1), \\ &\vec{A}^{(1,0)}_{1,0} = (2/5,1/5), \quad \vec{A}^{(1,0)}_{1,1} = (1/2,0), \\ &\vec{A}^{(0,0)}_{2,0} = (0,0), \qquad \vec{A}^{(0,0)}_{2,1} = (1,0), \qquad \vec{A}^{(0,0)}_{2,2} = (-1,1), \end{aligned}$$

$$\begin{aligned} &\vec{A}^{(2,0)}_{0,0} = (3/5,4/5), \\ &\vec{A}^{(1,0)}_{1,0} = (2/5,1/5), \quad \vec{A}^{(1,0)}_{1,1} = (1/2,0), \\ &\vec{A}^{(0,0)}_{2,0} = (0,0), \qquad \vec{A}^{(0,0)}_{2,1} = (1,0), \qquad \vec{A}^{(0,0)}_{2,2} = (-1,1), \end{aligned}$$

$$\begin{aligned} &\vec{A}^{(2,0)}_{0,0} = (3/5,4/5), \\ &\vec{A}^{(1,0)}_{1,0} = (2/5,1/5), \quad \vec{A}^{(1,1)}_{1,1} = (2,-4), \\ &\vec{A}^{(0,0)}_{2,0} = (0,0), \qquad \vec{A}^{(0,1)}_{2,1} = (1,0), \qquad \vec{A}^{(0,1)}_{2,2} = (-1,1), \end{aligned}$$

$$\begin{aligned} &\vec{A}^{(2,0)}_{0,0} = (3/5,4/5), \\ &\vec{A}^{(1,0)}_{1,0} = (2/5,1/5), \quad \vec{A}^{(1,1)}_{1,1} = (2,-4), \\ &\vec{A}^{(0,0)}_{2,0} = (0,0), \qquad \vec{A}^{(0,1)}_{2,1} = (1,0), \qquad \vec{A}^{(0,2)}_{2,2} = (-2/5,1/5). \end{aligned}$$

By Theorem 2.1

$$\begin{aligned}
\vec{S}_0(LB;y) &= \vec{A}^{(0,0)}_{2,0} + \frac{y-y_0|}{|\vec{A}^{(0,1)}_{2,1}} + \frac{y-y_1|}{|\vec{A}^{(0,2)}_{2,2}} \\
&= (0,0) + \frac{y\ |}{|(1,0)} + \frac{y-1\ |}{|(-2/5,1/5)} \\
&= \frac{(3y-2y^2, y^2-y)}{(3-2y)^2+(y-1)^2},
\end{aligned}$$

$$\begin{aligned}
\vec{S}_1(LB;y) &= \vec{A}^{(1,0)}_{1,0} + \frac{y-y_0|}{|\vec{A}^{(1,1)}_{1,1}} = (2/5,1/5) + \frac{y\ |}{|(2,-4)} \\
&= (\frac{4+y}{10}, \frac{1-y}{5}),
\end{aligned}$$

$$\vec{S}_2(LB;y) = \vec{A}^{(2,0)}_{0,0} = (3/5,4/5).$$

This leads to

$$\begin{aligned}
\vec{R}_2(LB;x,y) &= \vec{S}_0(LB;y) + \frac{x-x_2\ |}{|\vec{S}_1(LB;y)} + \frac{x-x_1\ |}{|\vec{S}_2(LB;y)} \\
&= \frac{(3y-2y^2, y^2-y)}{(3-2y)^2+(y-1)^2} + \frac{x+2\ |}{|(\frac{4+y}{10}, \frac{1-y}{5})} + \frac{x+1\ |}{|(3/5,4/5)} \\
&= \frac{(3y-2y^2, y^2-y)}{(3-2y)^2+(y-1)^2} + \frac{(10x+20)(6x+y+10, 8x-2y+10)}{(6x+y+10)^2+(8x-2y+10)^2}.
\end{aligned}$$

According to (2.16)–(2.18), one derives

$$\begin{aligned}
\vec{B}^{(0,0)}_{0,1} = (1,0), \quad & \vec{B}^{(0,0)}_{0,2} = (1,1), \\
& \vec{B}^{(1,0)}_{1,2} = (-1,0),
\end{aligned}$$

$$\begin{aligned}
\vec{B}^{(0,1)}_{0,1} = (1,0), \quad & \vec{B}^{(0,1)}_{0,2} = (1,1), \\
& \vec{B}^{(1,2)}_{1,2} = (-1,0),
\end{aligned}$$

$$\begin{aligned}
\vec{B}^{(0,1)}_{0,1} = (1,0), \quad & \vec{B}^{(0,2)}_{0,2} = (0,1), \\
& \vec{B}^{(1,2)}_{1,2} = (-1,0).
\end{aligned}$$

By Theorem 2.1

$$\vec{S}_0(RU;y) = \vec{B}^{(0,1)}_{0,1} + \frac{y-y_1|}{|\vec{B}^{(0,2)}_{0,2}} = (1,0) + \frac{y-1|}{|(0,1)} = (1, y-1),$$

$$\vec{S}_1(RU;y) = \vec{B}^{(1,2)}_{1,2} = (-1,0),$$

which results in

$$\begin{aligned}\vec{R}_2(RU;x,y) &= \vec{S}_0(RU;y) + \frac{x - x_0 \,|}{|\, \vec{S}_1(RU;y)} \\ &= (1, y-1) + \frac{x \,|}{|\, (-1,0)} = (1-x, y-1).\end{aligned}$$

Hence we finally obtain

$$\begin{aligned}\vec{R}_2(x,y) &= Q(x,y)\vec{R}_2(LB;x,y) + P(x,y)\vec{R}_2(RU;x,y) \\ &= (x+y-2)(x+y-1)\left[\frac{(3y-2y^2, y^2-y)}{(3-2y)^2+(y-1)^2}\right. \\ &\qquad\qquad \left. + \frac{(10x+20)(6x+y+10, 8x-2y+10)}{(6x+y+10)^2+(8x-2y+10)^2}\right] \\ &\quad + (x+y+2)(x+y+1)(x+y)(1-x, y-1).\end{aligned}$$

It is easy to verify that $\vec{R}_2(x,y)$ interpolates $\vec{V}^2$ over $\Pi^2$. In our example, the vector-grid $\vec{V}^2$ is ill-defined, and, what is more, as mentioned at the beginning of this section, in this case one fails to find a rational interpolant $\vec{r}_2(x,y)$ of the form (1.4). However, $\vec{R}_2(x,y)$, as a BCVRI defined in (2.11), still exists. Hence, compared with Algorithm 1.1, our new algorithm for BCVRI is more reliable in the sense that it can overcome the nonexistence of some $\vec{r}_n(x,y)$, and more economical in the sense that it involves fewer Samelson inverses.

## Acknowledgments

The authors would like to thank an anonymous referee for his or her valuable suggestions on the rewriting of Section 1, and for other helpful comments.

## References

[1] Cuyt A. and Verdonk B., *Multivariate rational interpolation*, Computing **34** (1985), 41–61. MR **86g:**41003

[2] ———, *A review of branched continued fraction theory for the construction of multivariate rational approximants*, Appl. Numer. Math. **4** (1988), 263–271. MR **89k:**41018

[3] Graves-Morris P.R., *Vector valued rational interpolants I*, Numer. Math. **42** (1983), 331–348. MR **85a:**65020

[4] Siemaszko W., *Thiele-type branched continued fractions for two-variable functions*, J. Comput. Appl. Math. **9** (1983), 137–153. MR **85h:**65024

[5] Zhu G.Q. and Tan J.Q., *The duality of bivariate vector valued rational interpolants over rectangular grids*, Math. Num. Sin. **3** (1995), 311–320; English transl., Chinese J. Numer. Math. Appl. **17** (1995), no. 4, 75–84. MR **98i:**41015

Institute of Applied Mathematics, Hefei University of Technology, Hefei 230009, P. R. China

*E-mail address*: jqtan@hfut.edu.cn