Egyptian Informatics Journal (2012) 13, 147–154

Cairo University

Egyptian Informatics Journal

www.elsevier.com/locate/eij
www.sciencedirect.com

ORIGINAL ARTICLE

Shape preserving rational bi-cubic function

Malik Zawwar Hussain [a], Maria Hussain [b,*], Madiha Amjad [a]

[a] Department of Mathematics, University of the Punjab, Lahore, Pakistan
[b] Lahore College for Women University, Lahore, Pakistan

Received 13 October 2011; revised 26 June 2012; accepted 28 June 2012
Available online 10 October 2012

KEYWORDS

Rational bi-cubic functions;
Parameters;
Monotone surface;
Convex surface

Abstract The study is dedicated to the development of shape preserving interpolation scheme for monotone and convex data. A rational bi-cubic function with parameters is used for interpolation. To preserve the shape of monotone and convex data, the simple data dependent constraints are developed on these parameters in each rectangular patch. The developed scheme of this paper is confined, cheap to run and produce smooth surfaces.

1. Introduction

Monotonicity is a key tool for the specification of Digital to Analog Converters (DACs), Analog to Digital Converters (ADCs) and sensors. These devices are enormously used in control system applications. Erythrocyte sedimentation rate (ESR) in cancer patients, uric acid level in patients suffering from gout, approximation of couples and quasi couples in statistics are a few other monotone quantities. Convexity arises in the data generated in nonlinear programming, scientific applications such as design, optimal control, parameter estimation and approximations.

A fine quantity of work [1–9] has been published in previous years that emphasized on the shape preservation of curves and surfaces. Beatson and Ziegler [2] developed a shape preserving interpolation scheme for regular monotone data. The rectangular patches were joined by C^1 quadratic spline. Sufficient conditions were derived on data values and derivatives arranged on the rectangular grid to ensure monotonicity. Carlson and Fritsch [3] developed a bi-cubic polynomial interpolation scheme for shape preservation of monotone data. The conditions were worked out on derivatives. These conditions were sufficient to establish monotone bi-cubic polynomial over all rectangular elements. Sarfraz, Butt and Hussain [8] constructed a rational interpolation scheme for monotone data. The scheme was cubic in each co-ordinate parameter. The rational interpolant had one free parameter in each subinterval. Monotonicity was assured by establishing automotive constraints on the free parameter.

Asaturyan [1] proposed a subdivision scheme for convex surface data interpolation. The rectangular patches in which convexity was lost were identified and divided into nine sub-rectangles. Convexity preserving interpolation scheme was applied on these sub-rectangles. It was observed that change in a sub-rectangle affected the whole domain thus established

* Corresponding author. Tel.: +92 0331 4930071.
E-mail addresses: malikzawwar.math@pu.edu.pk (M.Z. Hussain), mariahussain1@yahoo.com, mariahussain_1@yahoo.com (M. Hussain).
Peer review under responsibility of Faculty of Computers and Information, Cairo University.

ELSEVIER Production and hosting by Elsevier

1110-8665 © 2012 Faculty of Computers and Information, Cairo University. Production and hosting by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.eij.2012.06.001

it a global scheme. Costantini and Fontanella [4] developed an interpolation scheme for regular convex data. The scheme was semi global. Floater [5] derived sufficient conditions on the control points of Bézier surfaces to raise convex surface from convex data. Hussain, Sarfraz and Shaikh [7] developed a C^2 rational scheme for shape preservation of positive and convex data.

The paper underlines the problem of monotone and convex data interpolation. The data values are interpolated by rational bi-cubic function. This rational bi-cubic function enjoys eight parameters in each rectangular patch. The range of these parameters is determined to ensure shape preservation of data. The developed schemes of this paper deal positively to both data and data with derivatives, assure local command on the surface and promise C^1 smoothness.

The rest of the paper is organized as follows: The Section 2 reviews [9]. Section 3 details the rational bi-cubic function [6] used for surface data interpolation. The shape preserving interpolation schemes for monotone and convex data are developed in Section 4. Section 5 concludes the paper.

2. Rational cubic function

This section details the rational cubic function developed by Sarfraz and Hussain [9].

Let the planar data under consideration be $\{(x_i, f_i) : i = 0, 1, 2, \ldots, n\}$ and the partition of knot be $x_i < x_{i+1}$, $i = 0,\ 1,\ 2,\ \ldots,\ n-1$. The function and derivative values are f_i, d_i respectively. A piecewise C^1 rational cubic function is defined as:

$$S(x) \equiv S_i(x) = \frac{p_i(\theta)}{q_i(\theta)}, \tag{1}$$

where

$$p_i(\theta) = U_i(1-\theta)^3 + v_i V_i \theta(1-\theta)^2 + w_i W_i \theta^2(1-\theta) + Z_i\theta^3,$$

$$q_i(\theta) = (1-\theta)^3 + v_i\theta(1-\theta)^2 + w_i\theta^2(1-\theta) + \theta^3,$$

$$U_i = f_i,\ Z_i = f_{i+1},\ V_i = f_i + \frac{h_i d_i}{v_i},\ W_i = f_{i+1} - \frac{h_i d_{i+1}}{w_i},$$

$$\theta = \frac{x - x_i}{h_i},\ h_i = x_{i+1} - x_i,\ \Delta_i = \frac{f_{i+1} - f_i}{h_i}.$$

v_i and w_i are the free parameters, variations to the values of v_i's and w_i's help the user to control (tighten or loosen) the shape of the curve in different pieces. One can note that when, $v_i = w_i = 3$, then the rational cubic function obviously become cubic Hermite interpolant.

Theorem 1 [9]. For the given conditions $d_i \geqslant 0,\ i = 0,\ 1,\ 2,\ \ldots,\ n$ on the derivative parameters, the sufficient conditions for the interpolant (1) to be monotonically increasing are

$$v_i = \frac{d_i + d_{i+1}}{\Delta_i},\ w_i = \frac{d_i + d_{i+1}}{\Delta_i}.$$

Theorem 2 [9]. For the given conditions $d_1 < \Delta_1 < \cdots < \Delta_{i-1} < d_i < \Delta_i < \cdots < \Delta_{n-1} < d_n$ on the derivative parameters and data, the sufficient conditions for the interpolant (1) to be convex are

$$v_i = w_i = q_i + Max\left(\frac{d_{i+1} - d_i}{\Delta_i - d_i},\ \frac{d_{i+1} - d_i}{d_{i+1} - \Delta_i}\right);\ q_i = 0.$$

The proofs of Theorem 1 and Theorem 2 can found in [9].

3. Rational bi-cubic function

In this section, the extension of rational cubic function (1) to the rational bi-cubic function $S(x, y)$ for the interpolation of regular data is described. Let the regular data be arranged over the rectangular domain $D = [a, b] \times [c, d]$. The partition of the intervals $[a, b]$ and $[c, d]$ are $\pi : a = x_0 < x_1 < x_2 < \cdots < x_m = b$ and $\tilde{\pi} : c = y_0 < y_1 < y_2 < \cdots < y_n = d$. The representation of rational bi-cubic function over the rectangular patch $I = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$ is

$$\begin{aligned} S(x,y) = {} & a_0(\alpha)S(x_i, y) + a_1(\alpha)S(x_{i+1}, y) - S(x, y_j)b_0(\beta) \\ & - S(x, y_{j+1})b_1(\beta) \\ & - a_0(\alpha)\{b_0(\beta)S(x_i, y_j) + b_1(\beta)S(x_i, y_{j+1})\} \\ & - a_1(\alpha)\{b_0(\beta)S(x_{i+1}, y_j) + b_1(\beta)S(x_{i+1}, y_{j+1})\}, \end{aligned} \tag{2}$$

$a_i(\alpha)$ and $b_i(\beta)$, $i = 0,\ 1$ are the cubic Hermite blending functions defined as:

$$\begin{aligned} a_0(\alpha) &= (1-\alpha)^2(1+2\alpha),\ a_1(\alpha) = \alpha^2(3-2\alpha),\ b_0(\beta) \\ &= (1-\beta)^2(1+2\beta),\ b_1(\beta) = \beta^2(3-2\beta), \alpha = \frac{x - x_i}{\delta_i},\ \beta \\ &= \frac{y - y_j}{\hat{\delta}_j},\ i = 0, 1, 2, \cdots, m-1,\ j = 0, 1, 2, \cdots, n-1. \end{aligned}$$

$S(x, y_j)$, $S(x, y_{j+1})$, $S(x_i, y)$ and $S(x_{i+1}, y)$ are the rational cubic function (1) constituting the four boundaries of rectangular patch $I = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$ as:

$$S(x, y_j) = \frac{\sum_{i=0}^{3}(1-\alpha)^{3-i}\alpha^i \tau_{0,i}}{\chi_1(\alpha)}, \tag{3}$$

with

$$\begin{aligned} \tau_{0,0} &= F_{i,j},\ \tau_{0,1} = v_{i,j}F_{i,j} + \delta_i F^x_{i,j},\ \tau_{0,2} = w_{i,j}F_{i+1,j} - \delta_i F^x_{i+1,j},\ \tau_{0,3} \\ &= F_{i+1,j},\ \chi_1(\alpha) \\ &= (1-\alpha)^3 + v_{i,j}\alpha(1-\alpha)^2 + w_{i,j}\alpha^2(1-\alpha) + \alpha^3, \end{aligned}$$

$$S(x, y_{j+1}) = \frac{\sum_{i=0}^{3}(1-\alpha)^{3-i}\alpha^i \tau_{1,i}}{\chi_2(\alpha)}, \tag{4}$$

with

$$\begin{aligned} \tau_{1,0} &= F_{i,j+1},\ \tau_{1,1} = v_{i,j+1}F_{i,j+1} + \delta_i F^x_{i,j+1},\ \tau_{1,2} \\ &= w_{i,j+1}F_{i+1,j+1} - \delta_i F^x_{i+1,j+1},\ \tau_{1,3} = F_{i+1,j+1},\ \chi_2(\alpha) \\ &= (1-\alpha)^3 + v_{i,j+1}\alpha(1-\alpha)^2 + w_{i,j+1}\alpha^2(1-\alpha) + \alpha^3, \end{aligned}$$

$$S(x_i, y) = \frac{\sum_{i=0}^{3}(1-\beta)^{3-i}\beta^i \tau_{2,i}}{\chi_3(\beta)}, \tag{5}$$

with

$$\begin{aligned} \tau_{2,0} &= F_{i,j},\ \tau_{2,1} = \hat{v}_{i,j}F_{i,j} + \hat{\delta}_j F^y_{i,j},\ \tau_{2,2} = \hat{w}_{i,j}F_{i,j+1} - \hat{\delta}_j F^y_{i,j+1},\ \tau_{2,3} \\ &= F_{i,j+1},\ \chi_3(\beta) \\ &= (1-\beta)^3 + \hat{v}_{i,j}\beta(1-\beta)^2 + \hat{w}_{i,j}\beta^2(1-\beta) + \beta^3, \end{aligned}$$

$$S(x_{i+1}, y) = \frac{\sum_{i=0}^{3}(1-\beta)^{3-i}\beta^i\tau_{3,i}}{\chi_4(\beta)}, \tag{6}$$

with

$$\tau_{3,0} = F_{i+1,j},\ \tau_{3,1} = \hat{v}_{i+1,j}F_{i+1,j} + \hat{\delta}_j F^y_{i+1,j},\ \tau_{3,2} = \hat{w}_{i+1,j}F_{i+1,j+1} - \hat{\delta}_j F^y_{i+1,j+1},\ \tau_{3,3} = F_{i+1,j+1},\ \chi_4(\beta) = (1-\beta)^3 + \hat{v}_{i+1,j}\beta(1-\beta)^2 + \hat{w}_{i+1,j}\beta^2(1-\beta) + \beta^3.$$

3.1. Arithmetic mean method of derivative approximation [6]

In this paper the partial derivatives $F^x_{i,j}$ and $F^y_{i,j}$ are approximated by 'arithmetic mean method.' Arithmetic mean method of derivative approximation is a difference scheme which involves three neighbouring points for the computation of derivatives at each ijth corner of rectangular patch. The explicit formulae are as follows:

$$F^x_{0,j} = \Delta_{0,j} + (\Delta_{0,j} - \Delta_{1,j})\frac{\delta_0}{\delta_0+\delta_1},$$

$$F^x_{m,j} = \Delta_{m-1,j} + (\Delta_{m-1,j} - \Delta_{m-2,j})\frac{\delta_{m-1}}{\delta_{m-1}+\delta_{m-2}},$$

$$F^x_{i,j} = \frac{\Delta_{i,j} + \Delta_{i-1,j}}{2},\ i = 1,2,3,\cdots,m-1,\ j = 0,1,2,\cdots,n.$$

$$F^y_{i,0} = \widehat{\Delta}_{i,0} + (\widehat{\Delta}_{i,0} - \widehat{\Delta}_{i,1})\frac{\hat{\delta}_0}{\hat{\delta}_0+\hat{\delta}_1},$$

$$F^y_{i,n} = \widehat{\Delta}_{i,n-1} + (\widehat{\Delta}_{i,n-1} - \widehat{\Delta}_{i,n-2})\frac{\hat{\delta}_{n-1}}{\hat{\delta}_{n-1}+\hat{\delta}_{n-2}},$$

$$F^y_{i,j} = \frac{\widehat{\Delta}_{i,j} + \widehat{\Delta}_{i,j-1}}{2},\ i = 0,1,2,\cdots,m,\ j = 1,2,3,\cdots,n-1,$$

where $\Delta_{i,j} = \frac{F_{i+1,j}-F_{i,j}}{\delta_i}$ and $\widehat{\Delta}_{i,j} = \frac{F_{i,j+1}-F_{i,j}}{\hat{\delta}_j}$.

4. The proposed algorithms

In this section the monotonicity and convexity preserving schemes are developed for regular surface data using the rational bi-cubic function (2).

4.1. Monotone rational bi-cubic function

Theorem 3. The rational function defined in (2) preserves the shape of monotone data if in every rectangular patch, the free parameters $v_{i,j}$, $w_{i,j}$, $v_{i,j+1}$, $w_{i,j+1}$, $\hat{v}_{i,j}$, $\hat{w}_{i,j}$, $\hat{v}_{i+1,j}$, $\hat{w}_{i+1,j}$ satisfy the following conditions

$$v_{i,j} = w_{i,j} > Max\left\{0, \frac{F^x_{i,j} + F^x_{i+1,j}}{\Delta_{i,j}}\right\},$$

$$\hat{v}_{i,j} = \hat{w}_{i,j} > Max\left\{0, \frac{F^y_{i,j} + F^y_{i,j+1}}{\widehat{\Delta}_{i,j}}\right\},$$

$$v_{i,j+1} = w_{i,j+1} > Max\left\{0, \frac{F^x_{i,j+1} + F^x_{i+1,j+1}}{\Delta_{i,j+1}}\right\},$$

$$\hat{v}_{i+1,j} = \hat{w}_{i+1,j} > Max\left\{0, \frac{F^y_{i+1,j} + F^y_{i+1,j+1}}{\widehat{\Delta}_{i+1,j}}\right\}.$$

Proof. Let $\{(x_i, y_j, F_{i,j}) : i = 0,1,2,\ldots,m;\ j = 0,1,2,\ldots,n\}$ be the monotone regular data. The data is arranged over the rectangular region $I = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$, $i = 0,1,2,\ldots,m-1$, $j = 0,1,2,\ldots,n-1$. Since the data is monotone so $F^x_{i,j} > 0$, $F^y_{i,j} > 0$, $\Delta_{i,j} > 0$ and $\widehat{\Delta}_{i,j} > 0$, (the necessary conditions for monotonicity of data).

The monotonicity of rational function (2) is dependent on the monotonicity of boundary curves ((3)-(6)). The boundary curves are monotone if $S^{(1)}(x_l, y) > 0$ and $S^{(1)}(x, y_k) > 0$, $l = i, i+1$ and $k = j, j+1$. The complete expressions of $S^{(1)}(x_l, y) > 0$ and $S^{(1)}(x, y_k) > 0$ are as follows:

$$S^{(1)}(x, y_j) = \frac{\sum_{i=0}^{5}(1-\alpha)^{5-i}\alpha^i\tau_{4,i}}{(\chi_1(\alpha))^2}, \tag{7}$$

with

$$\tau_{4,0} = F^x_{i,j},\ \tau_{4,1} = 2(w_{i,j}\Delta_{i,j} - F^x_{i+1,j}) + F^x_{i,j},$$

$$\tau_{4,2} = \Delta_{i,j} + 2(w_{i,j}\Delta_{i,j} - F^x_{i+1,j}) + (v_{i,j}w_{i,j}\Delta_{i,j} - w_{i,j}F^x_{i,j} - v_{i,j}F^x_{i+1,j}),$$

$$\tau_{4,3} = 3\Delta_{i,j} + 2(w_{i,j}\Delta_{i,j} - F^x_{i,j}) + (v_{i,j}w_{i,j}\Delta_{i,j} - w_{i,j}F^x_{i,j} - v_{i,j}F^x_{i+1,j}),$$

$$\tau_{4,4} = 2(v_{i,j}\Delta_{i,j} - F^x_{i,j}) + F^x_{i+1,j},\ \tau_{4,5} = F^x_{i+1,j}.$$

The $S^{(1)}(x, y_{j+1})$ can be computed by replacing j by $j+1$ in (7).

The boundary curve $S(x, y_j)$ is monotone if $\sum_{i=0}^{5}(1-\alpha)^{5-i}\alpha^i\tau_{4,i} > 0$, which is true only if $\tau_{4,i}$'s are positive. $\tau_{4,i}$'s, $i = 0, 1, 2, 3, 4, 5$ are positive if the parameters $v_{i,j}$ and $w_{i,j}$ observe the following restrictions

$$v_{i,j} = w_{i,j} > Max\left\{0, \frac{F^x_{i,j} + F^x_{i+1,j}}{\Delta_{i,j}}\right\}$$

Similarly, the boundary curve $S(x, y_{j+1})$ is monotone if

$$v_{i,j+1} = w_{i,j+1} > Max\left\{0, \frac{F^x_{i,j+1} + F^x_{i+1,j+1}}{\Delta_{i,j+1}}\right\}.$$

The computations for monotonicity of boundary curves $S(x_l, y)$, $l = i, i+1$ are

$$S^{(1)}(x_i, y) = \frac{\sum_{i=0}^{5}(1-\beta)^{5-i}\beta^i\tau_{5,i}}{(\chi_3(\beta))^2}, \tag{8}$$

with

$$\tau_{5,0} = F^y_{i,j},\ \tau_{5,1} = 2(\hat{w}_{i,j}\widehat{\Delta}_{i,j} - F^y_{i,j+1}) + F^y_{i,j},$$

$$\tau_{5,2} = \widehat{\Delta}_{i,j} + 2(\hat{w}_{i,j}\widehat{\Delta}_{i,j} - F^y_{i,j+1}) + (\hat{v}_{i,j}\hat{w}_{i,j}\widehat{\Delta}_{i,j} - \hat{w}_{i,j}F^y_{i,j} - \hat{v}_{i,j}F^y_{i,j+1}),$$

$$\tau_{5,3} = 3\widehat{\Delta}_{i,j} + 2(\hat{w}_{i,j}\widehat{\Delta}_{i,j} - F^y_{i,j}) + (\hat{v}_{i,j}\hat{w}_{i,j}\widehat{\Delta}_{i,j} - \hat{w}_{i,j}F^y_{i,j} - \hat{v}_{i,j}F^y_{i,j+1}),$$

$$\tau_{5,4} = 2(\hat{v}_{i,j}\widehat{\Delta}_{i,j} - F^y_{i,j}) + F^y_{i,j+1},\ \tau_{5,5} = F^y_{i,j+1}.,$$

Similarly, for the expression of $S^{(1)}(x_{i+1}, y)$, replace i by $i+1$ in (8).

$S(x_i, y)$ is monotone if $\sum_{i=0}^{5}(1-\beta)^{5-i}\beta^i\tau_{5,i} > 0$, which is true only if $\tau_{5,i}$'s are positive. $\tau_{5,i}$'s, $i = 0, 1, 2, 3, 4, 5$ are positive if the parameters $\hat{v}_{i,j}$ and $\hat{w}_{i,j}$ observe the following restrictions

$$\hat{v}_{i,j} = \hat{w}_{i,j} > Max\left\{0, \frac{F^y_{i,j} + F^y_{i,j+1}}{\hat{\Delta}_{i,j}}\right\}.$$

Similarly, the boundary curve $S(x_{i+1}, y)$ is monotone if

$$\hat{v}_{i+1,j} = \hat{w}_{i+1,j} > Max\left\{0, \frac{F^y_{i+1,j} + F^y_{i+1,j+1}}{\hat{\Delta}_{i+1,j}}\right\}.$$

Algorithm 1. It is the algorithm to apply Theorem 3.

Step 1: Compute $(m+1)\times(n+1)$ monotone data points $\{(x_i, y_j, F_{i,j}) : i = 0, 1, 2, \ldots, m;\ j = 0, 1, 2, \ldots, n\}$.
Step 2: Approximate the derivatives $F^x_{i,j}$ and $F^y_{i,j}$ at knots using Section 3.1.
Step 3: Determine the values of free parameters $v_{i,j}, w_{i,j}, v_{i,j+1}, w_{i,j+1}, \hat{v}_{i,j}, \hat{w}_{i,j}, \hat{v}_{i+1,j}, \hat{w}_{i+1,j}$ by using Theorem 3.
Step 4: Insert the values of $F_{i,j}, F^x_{i,j}, F^y_{i,j}, i = 0, 1, 2, \ldots, m;\ j = 0, 1, 2, \ldots, n$ and $v_{i,j}, w_{i,j}, v_{i,j+1}, w_{i,j+1}, \hat{v}_{i,j}, \hat{w}_{i,j}, \hat{v}_{i+1,j}, \hat{w}_{i+1,j}, i = 0, 1, 2, \ldots, m-1;\ j = 0, 1, 2, \ldots, n-1$ in rational bi-cubic function (2).

4.1.1. Demonstration

In this section two monotone data sets are considered. These data sets are interpolated by bi-cubic Hermite (Figs. 1–3 and 7–9) and monotonicity preserving scheme (Figs. 4–6 and 10–12) developed in Section 4.1. It is observed that bi-cubic Hermite although C^1 is unable to preserve the monotone shape of data, whereas, the monotonicity preserving scheme of Section 4.1 preserves the shape of data.

Example 1. The surface in Fig. 1 represents the bi-cubic Hermite interpolation of monotone data considered in Table 1.

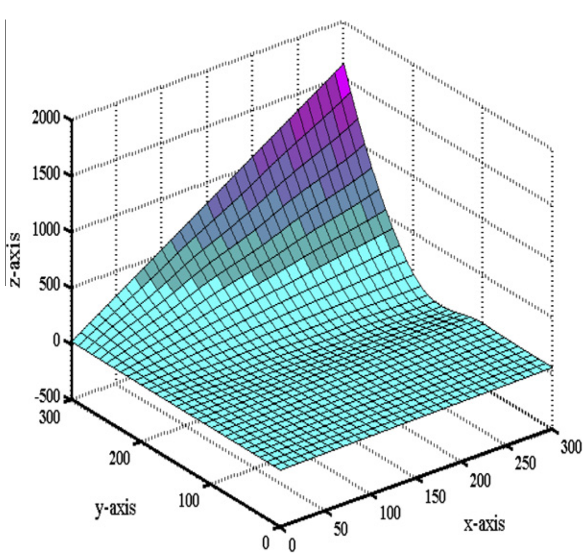

Figure 1 Bi-cubic Hermite interpolation.

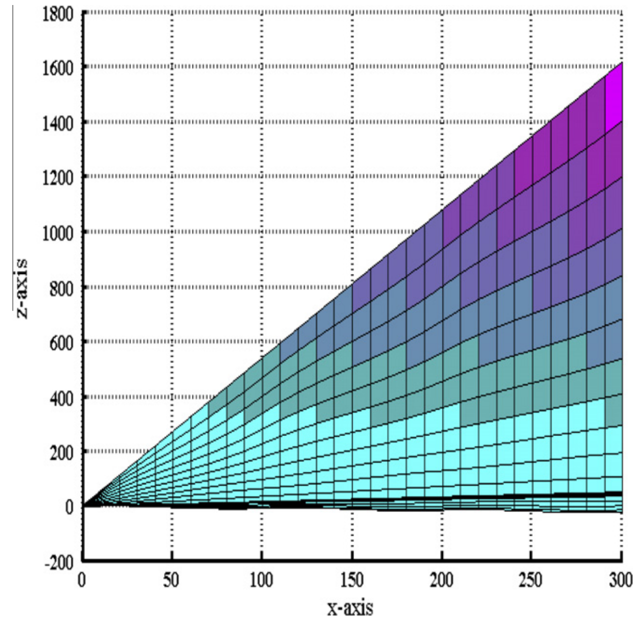

Figure 2 xz-view of Fig. 1.

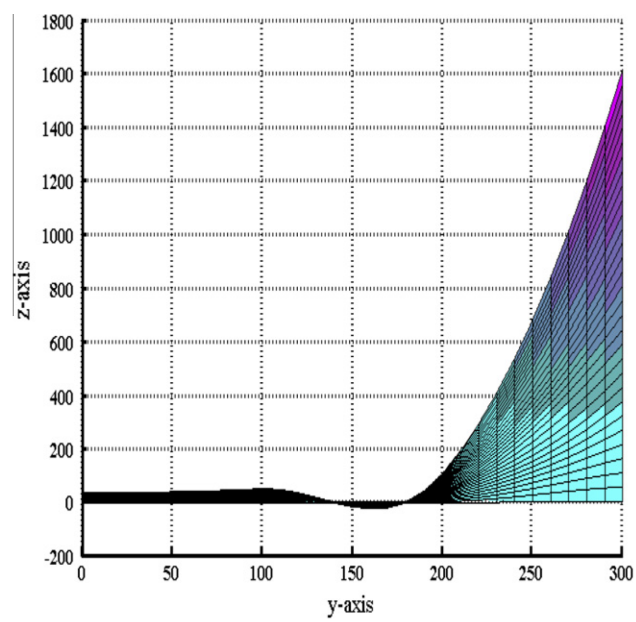

Figure 3 yz-view of Fig. 1.

The xz-view and yz-view of this figure is provided in Figs. 2 and 3 respectively. It is clear from Fig. 3 that the bi-cubic Hermite loses the monotone shape of data for $y \in [100, 200]$. The same data set is interpolated in Fig. 4 by C^1 monotone rational bi-cubic function developed in Section 4.1. Figs. 4–6 establish that monotonicity preserving scheme (developed in Section 4.1) works well.

Example 2. The Table 2 is of 3D data set, monotone over the whole domain.

The monotone data set in Table 2 is interpolated by bi-cubic Hermite in Fig. 7 which loses the monotone shape of data. It is more clear in Figs. 8 and 9 which provide the xz and yz views of

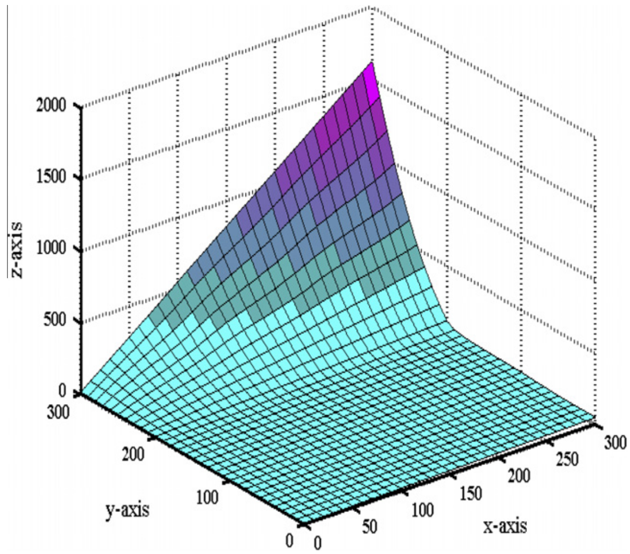

Figure 4 C^1 monotone rational bi-cubic function.

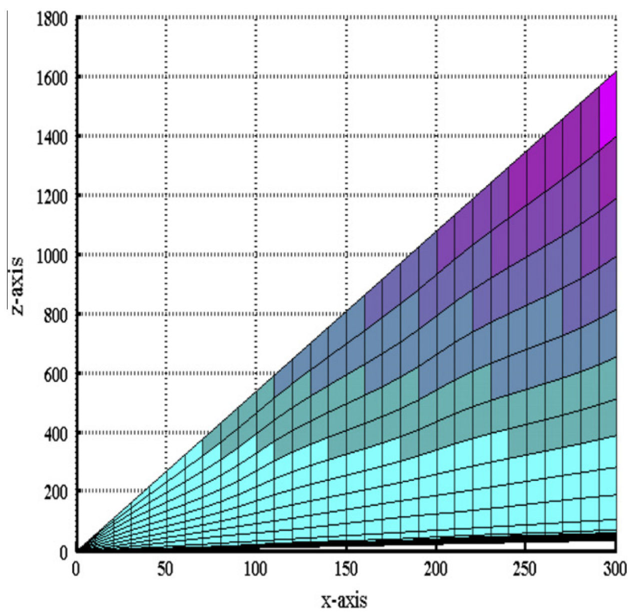

Figure 5 *xz*-view of Fig. 4.

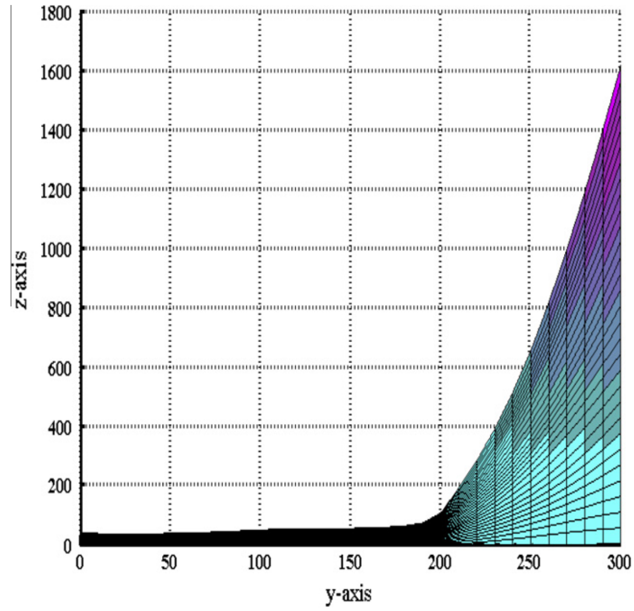

Figure 6 *yz*-view of Fig. 4.

Table 1 A 3D monotone data set.

y/x	1	100	200	300
1	0.1354	0.1744	0.3632	5.3814
100	13.5400	17.4370	36.3230	538.1400
200	27.0800	34.8730	72.6450	1076.3000
300	40.6200	52.3100	108.9700	1614.4000

Table 2 A 3D monotone data set.

y/x	0.1	0.3	0.7	1.3
0.1	0.0570	0.0572	0.0579	0.0593
0.3	0.0795	0.0801	0.0829	0.0869
0.7	0.6932	0.1451	0.1601	0.1715
1.3	0.1382	13.8160	15.8950	17.1111

same figure. Fig. 10 is the interpolation of same data by C^1 monotone rational bi-cubic function. It is comprehensible from Fig. 10 that C^1 monotone rational bi-cubic function developed in Section 4.1 preserved the monotone shape of data.

4.2. Convex rational bi-cubic function

Theorem 4. The bi-cubic function defined in (2) preserves the convexity if in each rectangular patch $I = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$ parameters $v_{i,j}, w_{i,j}, v_{i,j+1}, w_{i,j+1}, \hat{v}_{i,j}, \hat{w}_{i,j}, \hat{v}_{i+1,j}, \hat{w}_{i+1,j}$ satisfy the following conditions

$$v_{i,j} = w_{i,j} > Max\left\{0, \frac{F^x_{i+1,j} - F^x_{i,j}}{\Delta_{i,j} - F^x_{i,j}}, \frac{F^x_{i+1,j} - F^x_{i,j}}{F^x_{i+1,j} - \Delta_{i,j}}\right\},$$

$$\hat{v}_{i,j} = \hat{w}_{i,j} > Max\left\{0, \frac{F^y_{i,j+1} - F^y_{i,j}}{\widehat{\Delta}_{i,j} - F^y_{i,j}}, \frac{F^y_{i,j+1} - F^y_{i,j}}{F^y_{i,j+1} - \widehat{\Delta}_{i,j}}\right\},$$

$$v_{i,j+1} = w_{i,j+1} > Max\left\{0, \frac{F^x_{i+1,j+1} - F^x_{i,j+1}}{\Delta_{i,j+1} - F^x_{i,j+1}}, \frac{F^x_{i+1,j+1} - F^x_{i,j+1}}{F^x_{i+1,j+1} - \Delta_{i,j+1}}\right\},$$

$$\hat{v}_{i+1,j} = \hat{w}_{i+1,j} > Max\left\{0, \frac{F^y_{i+1,j+1} - F^y_{i+1,j}}{\widehat{\Delta}_{i+1,j} - F^y_{i+1,j}}, \frac{F^y_{i+1,j+1} - F^y_{i+1,j}}{F^y_{i+1,j+1} - \widehat{\Delta}_{i+1,j}}\right\}.$$

Proof. Let $\{(x_i, y_j, F_{i,j}):\ i = 0, 1, 2, \ldots, m;\ j = 0, 1, 2, \ldots, n\}$ be the given data arranged over the rectangular grid and satisfy the following necessary conditions of convexity

$$\Delta_{i,j} \leqslant \Delta_{i+1,j},\ \widehat{\Delta}_{i,j} \leqslant \widehat{\Delta}_{i,j+1},\ F^x_{i,j} \leqslant F^x_{i+1,j},\ F^y_{i,j} \leqslant F^y_{i,j+1},\ F^x_{i,j} \leqslant \Delta_{i,j} \leqslant F^x_{i+1,j},\ F^y_{i,j} \leqslant \widehat{\Delta}_{i,j} \leqslant F^y_{i,j+1}. \tag{9}$$

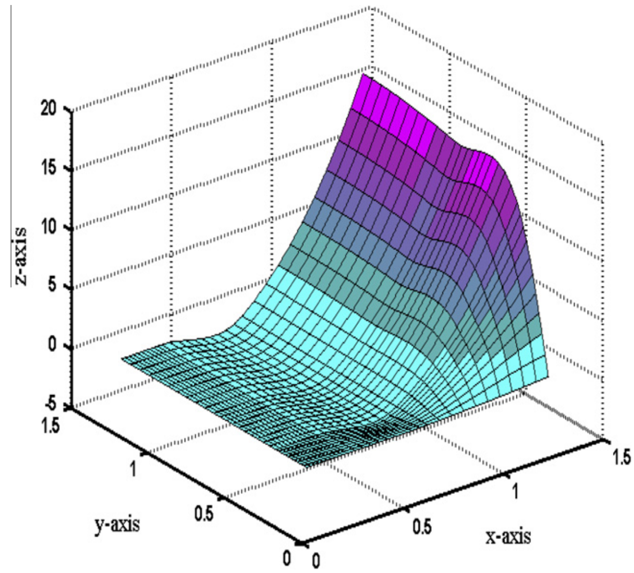

Figure 7 Bi-cubic Hermite interpolation.

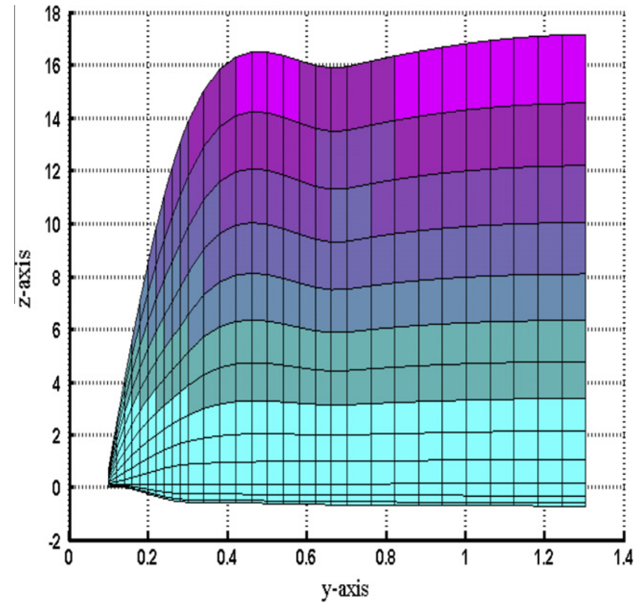

Figure 9 *yz*-view of Fig. 7.

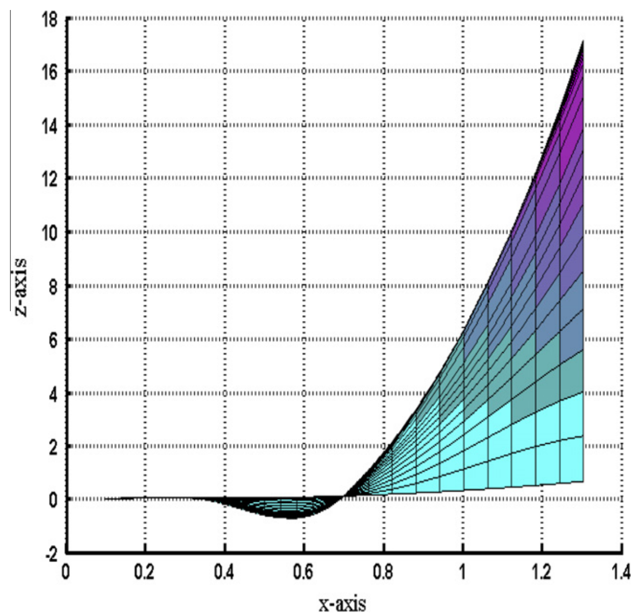

Figure 8 *xz*-view of Fig. 7.

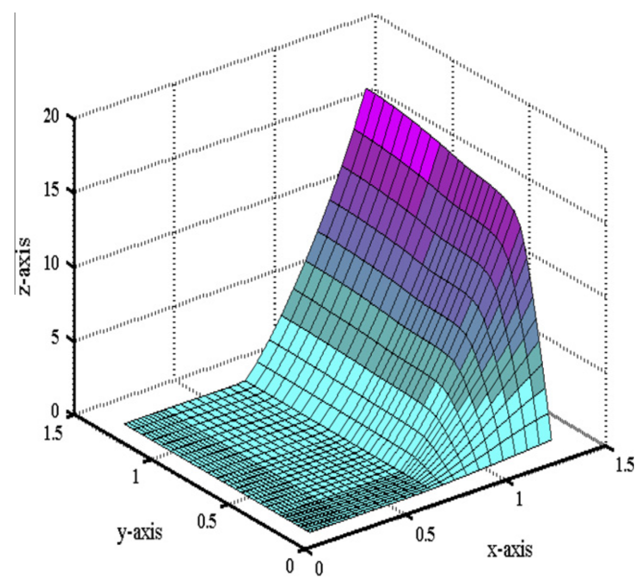

Figure 10 C^1 monotone rational bi-cubic function.

over each rectangular patch $I = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$, $i = 0, 1, 2, \ldots, m-1$; $j = 0, 1, 2, \ldots, n-1$.

The rational function (2) interpolates convex data as convex surface for convex boundary curves ((3)–(6)). The convex boundary curves are obtained by computing the constraints on the parameters $(v_{i,j}, w_{i,j}, v_{i,j+1}, w_{i,j+1}, \hat{v}_{i,j}, \hat{w}_{i,j}, \hat{v}_{i+1,j}, \hat{w}_{i+1,j})$ in the following way.

The boundary curves $S(x_l, y)$ and $S(x, y_k)$, $l = i, i+1$; $k = j, j+1$ are convex if $S^{(2)}(x_l, y) > 0$ and $S^{(2)}(x, y_k) > 0$, $l = i, i+1$; $k = j, j+1$.

$$S^{(2)}(x, y_j) = \frac{\sum_{i=0}^{7}(1-\alpha)^{7-i}\alpha^i\tau_{6,i}}{\delta_i(\chi_1(\alpha))^3}, \tag{10}$$

with

$$\tau_{6,0} = \tau_{4,1} - (2v_{i,j} - 1)\tau_{4,0},$$

$$\tau_{6,1} = 2\tau_{4,2} - (v_{i,j} - 2)\tau_{4,1} - (v_{i,j} + 4w_{i,j})\tau_{4,0},$$

$$\tau_{6,2} = 3(\tau_{4,3} + \tau_{4,2}) - 3w_{i,j}\tau_{4,1} - 3(w_{i,j} + 2)\tau_{4,0},$$

$$\tau_{6,3} = 4\tau_{4,4} + 4(v_{i,j} + 1)\tau_{4,3} + (v_{i,j} - 2w_{i,j})\tau_{4,2} - (2w_{i,j} + 5)\tau_{4,1} - 5\tau_{4,0},$$

$$\tau_{6,4} = 5\tau_{4,5} + (2v_{i,j} + 5)\tau_{4,4} + (2v_{i,j} - w_{i,j})\tau_{4,3} - (w_{i,j} - 4)\tau_{4,2} - 4\tau_{4,1},$$

$$\tau_{6,5} = 3(v_{i,j} + 2)\tau_{4,5} + 3v_{i,j}\tau_{4,4} - 3(\tau_{4,3} + \tau_{4,2}),$$

$$\tau_{6,6} = (4v_{i,j} + w_{i,j})\tau_{4,5} + (w_{i,j} - 2)\tau_{4,4} - 2\tau_{4,3},$$

$$\tau_{6,7} = (2w_{i,j} - 1)\tau_{4,5} - \tau_{4,4}.$$

The $S^{(2)}(x, y_{j+1})$ can be obtained by replacing j by $j + 1$ in (10).

The boundary curve $S(x, y_j)$ is convex if $\sum_{i=0}^{7}(1-\theta)^{7-i}\theta^i\tau_{6,i} > 0$. $\sum_{i=0}^{7}(1-\theta)^{7-i}\theta^i\tau_{6,i} > 0$ if $\tau_{6,i} > 0$, $i = 0, 1, 2, \ldots, 7$.

$\tau_{6,i} > 0$, $i = 0, 1, 2, \ldots, 7$ if

$$v_{i,j} = w_{i,j} > Max\left\{0, \frac{F^x_{i+1,j} - F^x_{i,j}}{\Delta_{i,j} - F^x_{i,j}}, \frac{F^x_{i+1,j} - F^x_{i,j}}{F^x_{i+1,j} - \Delta_{i,j}}\right\}.$$

Similarly, it can established that the boundary curve $S(x, y_{j+1})$ is convex if the parameters $v_{i,j+1}$ and $w_{i,j+1}$ satisfy the following constraints

$$v_{i,j+1} = w_{i,j+1} > Max\left\{0, \frac{F^x_{i+1,j+1} - F^x_{i,j+1}}{\Delta_{i,j+1} - F^x_{i,j+1}}, \frac{F^x_{i+1,j+1} - F^x_{i,j+1}}{F^x_{i+1,j+1} - \Delta_{i,j+1}}\right\}.$$

In the same way, for the convexity of other two vertical boundary curves, we have

$$S^{(2)}(x_i, y) = \frac{\sum_{i=0}^{7}(1-\beta)^{7-i}\beta^i\tau_{7,i}}{\hat{\delta}_i(\chi_3(\beta))^3}, \tag{11}$$

Table 3 A 3D convex data set.

y/x	-3	-1	0	1	3
-3	90	82	81	82	90
-1	10	2	1	2	10
0	9	1	0	1	9
1	10	2	1	2	10
3	90	82	81	82	90

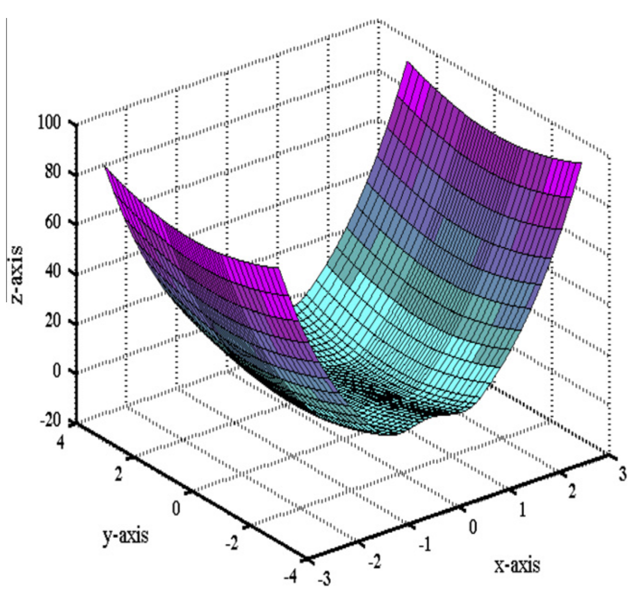

Figure 11 Bi-cubic Hermite interpolation.

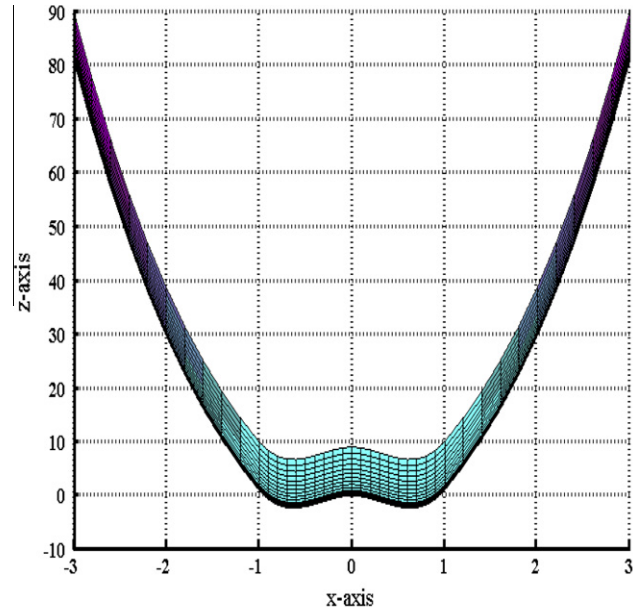

Figure 12 xz-view of Figure 11.

with

$$\tau_{7,0} = \tau_{5,1} - (2\hat{v}_{i,j} - 1)\tau_{5,0},$$

$$\tau_{7,1} = 2\tau_{5,2} - (\hat{v}_{i,j} - 2)\tau_{5,1} - (\hat{v}_{i,j} + 4\hat{w}_{i,j})\tau_{5,0},$$

$$\tau_{7,2} = 3(\tau_{5,3} + \tau_{5,2}) - 3\hat{w}_{i,j}\tau_{5,1} - 3(\hat{w}_{i,j} + 2)\tau_{5,0},$$

$$\tau_{7,3} = 4\tau_{5,4} + 4(\hat{v}_{i,j} + 1)\tau_{5,3} + (\hat{v}_{i,j} - 2\hat{w}_{i,j})\tau_{5,2} - (2\hat{w}_{i,j} + 5)\tau_{5,1} - 5\tau_{5,0},$$

$$\tau_{7,4} = 5\tau_{5,5} + (2\hat{v}_{i,j} + 5)\tau_{5,4} + (2\hat{v}_{i,j} - \hat{w}_{i,j})\tau_{5,3} - (\hat{w}_{i,j} - 4)\tau_{5,2} - 4\tau_{5,1},$$

$$\tau_{7,5} = 3(\hat{v}_{i,j} + 2)\tau_{5,5} + 3\hat{v}_{i,j}\tau_{5,4} - 3(\tau_{5,3} + \tau_{5,2}),$$

$$\tau_{7,6} = (4\hat{v}_{i,j} + \hat{w}_{i,j})\tau_{5,5} + (\hat{w}_{i,j} - 2)\tau_{5,4} - 2\tau_{5,3},$$

$$\tau_{7,7} = (2\hat{w}_{i,j} - 1)\tau_{5,5} - \tau_{5,4}.$$

The boundary curve $S(x_i, y)$ is convex if $\tau_{7,i} > 0$, $i = 0, 1, 2, \ldots, 7$. $\tau_{7,i} > 0$, $i = 0, 1, 2, \ldots, 7$ if

$$\hat{v}_{i,j} = \hat{w}_{i,j} > Max\left\{0, \frac{F^y_{i,j+1} - F^y_{i,j}}{\hat{\Delta}_{i,j} - F^y_{i,j}}, \frac{F^y_{i,j+1} - F^y_{i,j}}{F^y_{i,j+1} - \hat{\Delta}_{i,j}}\right\}.$$

The boundary curve $S(x_{i+1}, y)$ is convex if

$$\hat{v}_{i+1,j} = \hat{w}_{i+1,j} > Max\left\{0, \frac{F^y_{i+1,j+1} - F^y_{i+1,j}}{\hat{\Delta}_{i+1,j} - F^y_{i+1,j}}, \frac{F^y_{i+1,j+1} - F^y_{i+1,j}}{F^y_{i+1,j+1} - \hat{\Delta}_{i+1,j}}\right\}.$$

The above discussion can be wrapped up as:

Algorithm 2. It is the algorithm to apply Theorem 4.

Step 1. Compute $(m + 1) \times (n + 1)$ convex data points $\{(x_i, y_j, F_{i,j}) : i = 0, 1, 2, \ldots, m; j = 0, 1, 2, \ldots, n\}$.
Step 2. Approximate the derivatives $F^x_{i,j}$ and $F^y_{i,j}$ at knots using Section 3.1.

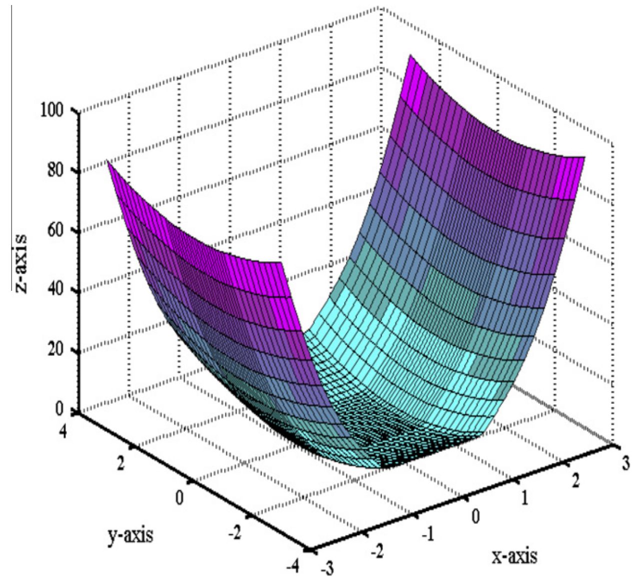

Figure 13 C^1 convex rational bi-cubic function.

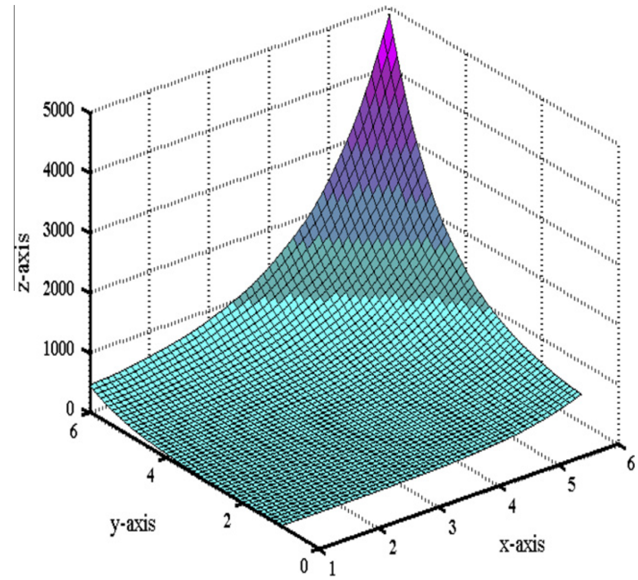

Figure 14 C^1 convex rational bi-cubic function.

Table 4 A 3D convex data set.

y/x	1	2	3	4	5	6
1	4.1	9.4	23.6	61.8	163.9	438.2
2	9.4	16.9	36.8	87.5	218.1	558.1
3	23.6	36.8	69.6	148.4	340.7	819.1
4	61.8	87.5	148.4	286.2	603.7	1354.4
5	163.9	218.1	340.7	603.7	1177.4	2465.7
6	438.2	558.1	819.1	1354.4	2465.7	4843.0

Step 3. Determine the values of the parameters $v_{i,j}$, $w_{i,j}$, $v_{i,j+1}$, $w_{i,j+1}$, $\hat{v}_{i,j}$, $\hat{w}_{i,j}$, $\hat{v}_{i+1,j}$, $\hat{w}_{i+1,j}$ by using Theorem 4.

Step 4. Insert the values of $F_{i,j}$, $F^x_{i,j}F^y_{i,j}$, $i = 0, 1, 2, \ldots, m$; $j = 0, 1, 2, \ldots, n$ $v_{i,j}$, $w_{i,j}$, $v_{i,j+1}$, $w_{i,j+1}$, $\hat{v}_{i,j}$, $\hat{w}_{i,j}$, $\hat{v}_{i+1,j}$, $\hat{w}_{i+1,j}$ $i = 0, 1, 2, \ldots, m$; $j = 0, 1, 2, \ldots, n$ in rational bi-cubic function (2).

4.2.1. Demonstration

Example 3. The convex data in Table 3 is computed from the convex function $F(x, y) = x^4 + y^2$. The domain is restricted to $[-3, 3] \times [-3, 3]$.

The convex data in Table 3 is interpolated by bi-cubic Hermite (Figs. 11 and 12). It is clear from these figures that bi-cubic Hermite loses convexity for $x \in [-1, 1]$. Hence bi-cubic Hermite is unable to preserve the convex shape of data. The convex surface in Fig. 13 is computed by interpolating the same data by convex interpolation scheme of Section 4.2.

Example 4. The convex data in Table 4 is of convex function $F(x, y) = e^{\sqrt{x^2+y^2}}$. The domain is restricted to $[1, 6] \times [1, 6]$.

The convex surface in Fig. 14 is the interpolation of convex data of Table 4. The interpolation is performed by the convexity preserving scheme of Section 4.2.

5. Conclusion

The paper deals with the problem of shape preservation of surface data. The rational bi-cubic function [6] can proficiently interpolate the data organize over rectangular grids. The monotonicity and convexity preservation of data is assured by computing constraints on free parameters. All these constraints involve data and derivatives. The shape preserving interpolation schemes developed in this paper works well for data with derivative; maintains smoothness and are local. The shape preserving interpolation schemes for convex [5] and monotone [2–4] data are futile for data with derivatives. The monotone data interpolation scheme [8] does enable derivative specification, but fails to maintain smoothness of surface. Moreover, the convex data interpolation schemes developed in [1,4] are global.

References

[1] Asaturyan, S., 1990. Shape preserving surface interpolation scheme, Ph.D. Thesis, Scotland (UK): Department of Mathematics and Computer Science, University of Dundee; 1990.
[2] Beatson RK, Ziegler Z. Monotonicity preserving surface interpolation. SIAM J Numer Anal 1985;22(2):401–11.
[3] Carlson RE, Fritsch FN. Monotone piecewise bicubic interpolation. SIAM J Numer Anal 1985;22(2):386–400.
[4] Costantini P, Fontanella F. Shape preserving bivariate interpolation. SIAM J Numer Anal 1990;27(2):488–506.
[5] Floater MS. A weak condition for the convexity of tensor-product Bézier and B-spline surfaces. Adv Comput Math 1994;2(1):67–80.
[6] Hussain MZ, Sarfraz M, Misbah Irshad, Madeha Amjad. Rational bi-cubic functions preserving 3D positive data. In: 8th International conference, computer graphics, imaging and visualization, Singapore; 16–19 August, 2011. p. 47–52.
[7] Hussain MZ, Sarfraz M, Shaikh TS. Shape preserving rational cubic spline for positive and convex data. Egyptian Inform J 2011;12(3): 231–6.
[8] Sarfraz M, Butt S, Hussain MZ. Surfaces for the visualization of scientific data preserving monotonicity. In: Goodman TNT, Martin R, editors. Proceedings of the IMA mathematics for surfaces VII, September 2–5, 1997, Dundee, UK; 1997. p. 479–95.
[9] Sarfraz M, Hussain MZ. Data visualization using rational spline interpolation. J Comput Appl Math 2006;189:513–25.