Spline Method Optimization of Bidimensional Functions

Liliana
Informatics Department, Faculty of Industrial Technology, Petra Christian University
Surabaya, Indonesia
lilian@petra.ac.id

Abstract- **A given set of scattered data usually need to be expressed in a function form. It is difficult to do because the scattered data may not similar with any known function, such as polynomial or trigonometric function. Spline function is a piecewise polynomial function which can approximate the scattered data better than usual polynomial function. Not only easy to construct and have good properties in avoiding big error when approximate a set of values, spline function also construct a smooth curve among the scattered data. B-spline is a specific spline with certain smoothness and degree, and domain partition. By using b-spline the approximation will be better and easy to construct.**

In this paper, it doesn't use a set of scattered data, but a set of values generated using a certain function. The experimental result shows that the spline can approximate the original function smoothly. As comparison to the set of values generated by a certain function, it is given a set of random value. The result also shows that spline approximates the random value well.

Keywords Spline, b-spline, approximation, bidimensional, piecewise function, optimization.

I. INTRODUCTION

For a given set of random points, they want to be represented in a function form or to be drawn on a representative curve. It's difficult to express the random values into a function. Manually, it can try various functions and compare with the values. The most similar function will be selected. This method actually is not a good and fast method. So, approximation theory is an important method indeed the manual method.

At the first time, it uses a polynomial and trigonometric basis to approximate the random values. Using the polynomial has a weakness, when the higher degree polynomial applied, the error will increase [1][2]. To avoid this kind of error, the better method is using spline.

Spline approximation is a piecewise polynomial function. If the spline is limited on a certain degree and smoothness, it called b-spline. This b-spline will approximate a part of the set of data using low degree polynomial.

In this project will be proved that b-spline method produces an almost-like smooth curve which approximate the data generated from a function.

II. SPLINES AND B-SPLINE

Spline is a piecewise polynomial function that differentiable up to a prescribe order [3]. The simplest spline is linear piecewise, spline order zero. The cubic spline will be called spline order two. Spline which written as an affine combination of some control points is named b-spline. B- spline can formed by joining polynomials together at fixed points called knots (control points)[1][2]. The control points will arrange the curve smoothness [4].

A b-spline has a certain degree, smoothness and domain partition [5][6][7]. A fundamental theorem states that every spline function of a given degree, smoothness and domain partitions, can be represented as a linear combination of B-splines of that same degree and smoothness, and over that same partition [1].

To make the approximation curve over all data, first, make the basis cubic spline, using equation 1. After that, construct the integration of four basis spline by shifting the basis spline on -1,1 and 2, as shown on figure 1.

$$B_3(x)=\begin{cases} 0 & , \ x\geq 2 \\ (2-x)^3/6 & ,1\leq x<2 \\ (1+3(1-x)+3(1-x)^2-3(1-x)^3)/6 & ,0\leq x<1 \\ B_3(-x) & , \ x<0 \end{cases} \quad (1)$$

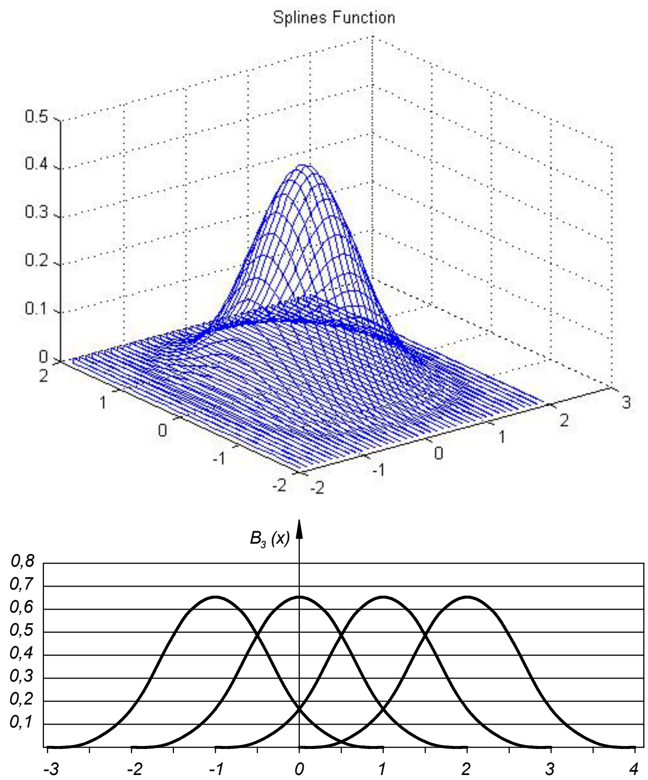

Figure 1. The basis function of cubic b-spline(above) and the integration splines (below)

Counting the basic factors using equation 2, the cubic b-spline can be obatain using equation 3. The last step is create the bidimensional spline using equation 4.

$$b_i = (1/6)(-f_{i-1} + 8f_i - f_{i+1}); \tag{2}$$

$$f(x) \cong S_m(x) = \sum_{i=-1}^{m+1} b_i \cdot B_i(x), \quad a \le x \le b \tag{3}$$

$$S_m(x, y) = \sum \sum b_{ij} B_{m,i}(x) B_{m,j}(y), \tag{4}$$

III. ALGORITHM

The main process is shown in figure 2. First, it needs the original function as input. Then, from the original function input, are taken 26 points as knots. The amount of knots is depended on the amount of basic factors (basic factors+2). The two extended points is because of the formula of finding basic factors as shown in eq 2.

Second, takes 40 points of basis of b-spline function and 24 basic factors. The amount of the basic factors is depended on how many approximated b-spline points wanted (approximated b-spline + 4). The four extended points are because of the formula, see eq 3. The last step is making the bidimensional b-spline using the approximate points.

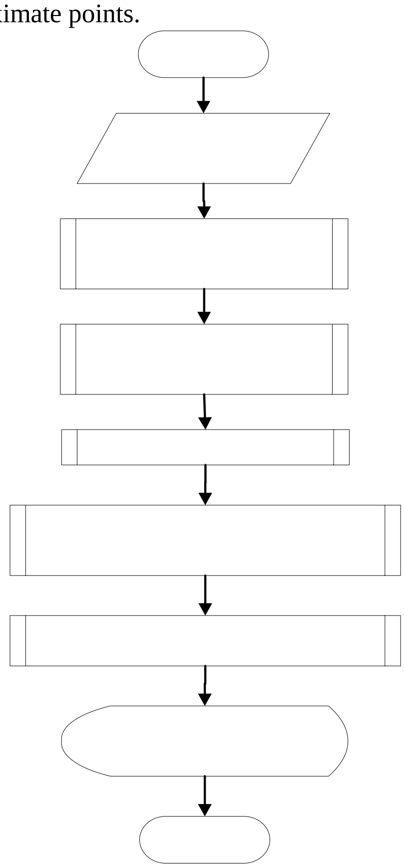

Figure 2. Flowchart of the main process

The more detail of each procedure will be explained through flowcharts on figure 3,4,5,6, and 7.

In making 24 b factors, actually it uses equation 2. Since equation 2 wants F[-1] and in program it can not use -1 for indexing, so, the 26 point of approximated function start from -0.1 (one point before 0) And reformulate the equation 2 become formula shown in figure 5.

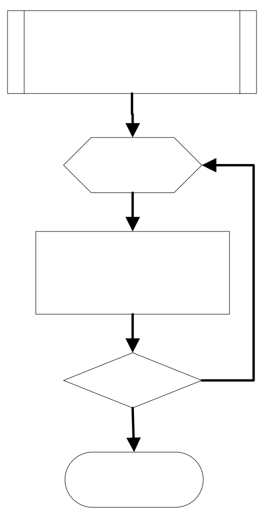

Figure 3. Flowchart of getting points from approximated function procedure

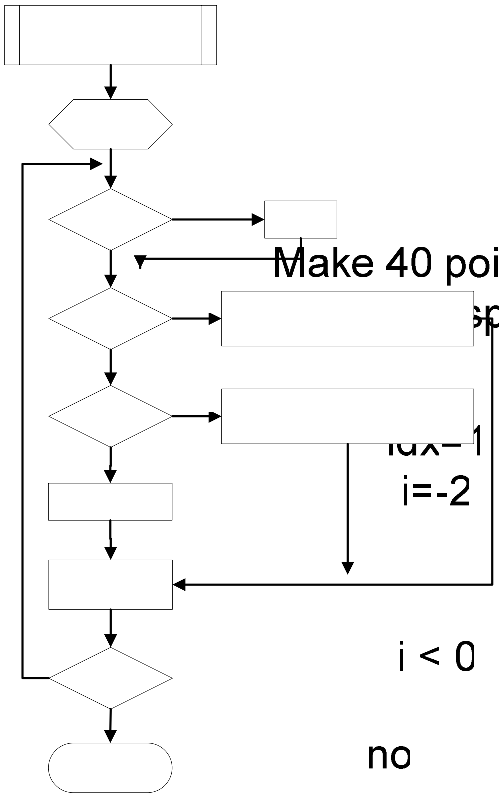

Figure 4. Flowchart of getting the basis function of spline procedure

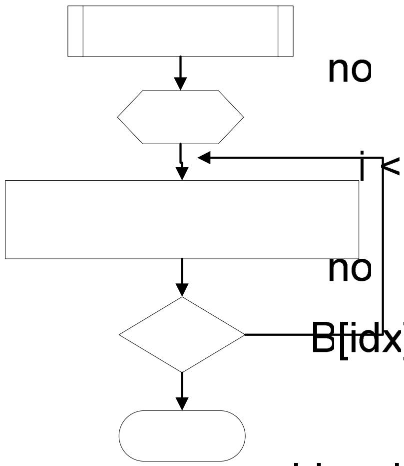

Figure 5. Flowchart of getting the b-factors procedure

In the procedure to get the 20 points, it is divided into two parts. First part, contains ten points, uses combination of four basis b-spline, in range 0 until 1. The second part, contains next ten points, just uses the combination of 3 basis splines.

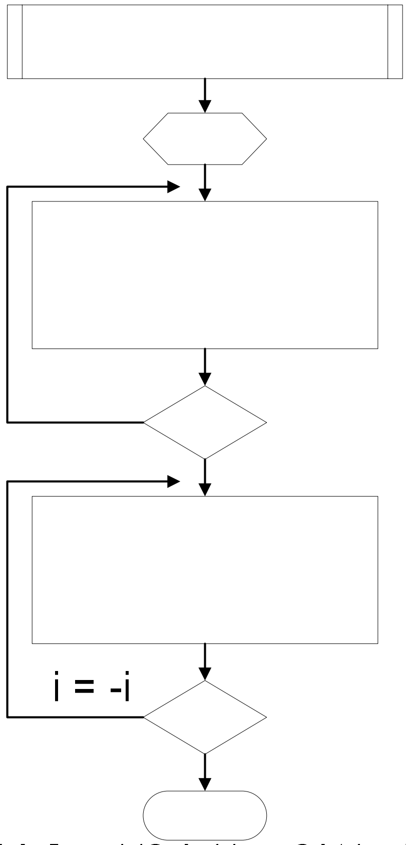

Figure 6. Flowchart of making the approximated points using basic spline

To make the entire bidimensional spline, firstly, generates the value and keeps in a 2D array. Then, plot the 2D array value. Generating the bidimensional spline uses the eq 4, but, simply it can be just multiplying the crossing index of two 1D splines.

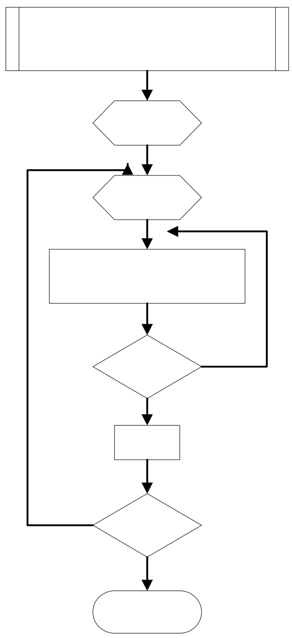

Figure 7. Flowchart of making the bidimensional spline

IV. EXPERIMENTAL RESULT

Figure 8 shows the one side of the original function and its approximation using b-spline. The bidimensional form shown in figure 9. Figure 10 shows the comparison between the original function and the b-splines approximation to the original function.

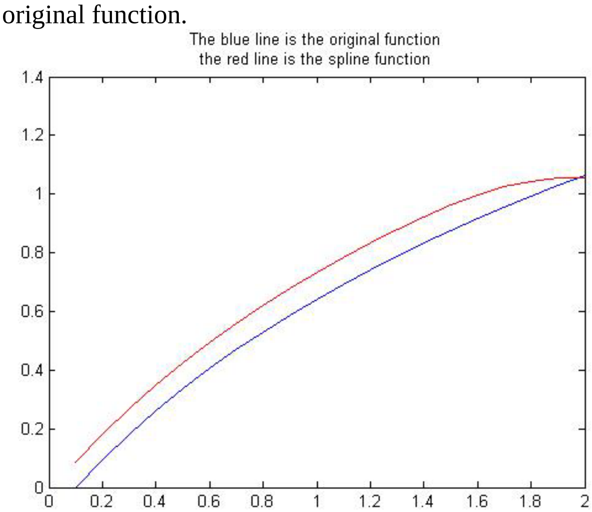

Figure 8. the original and approximated curve in one side

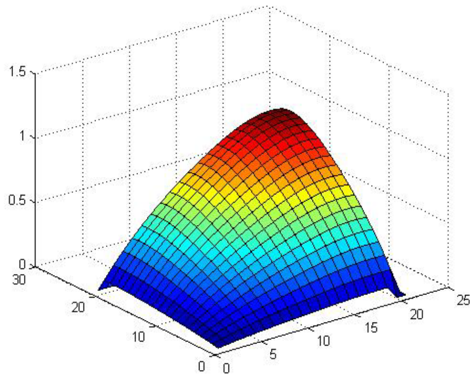

Figure 9. The spline approximation

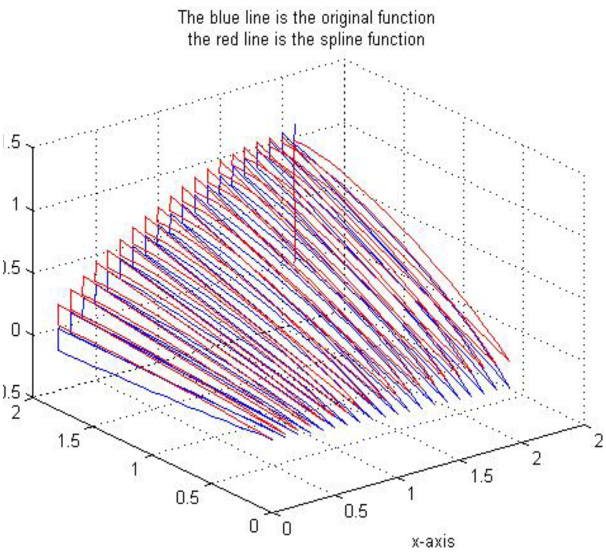

Figure 10. The original and b-spline curve

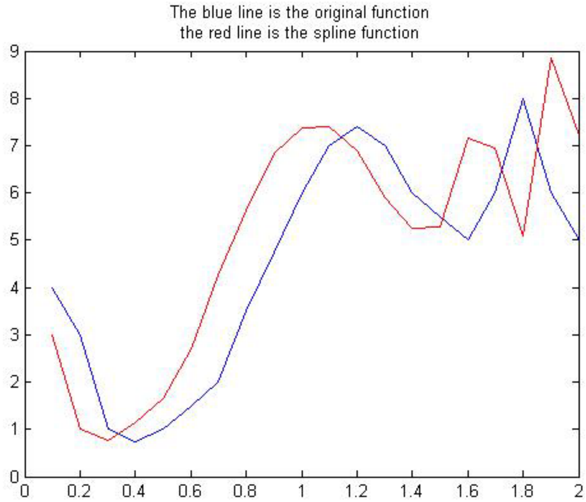

Figure 11. Spline approximation using random points

V. CONCLUSION

Using spline method, it can get an almost-like approximate curve to the original curve. If random values are used, the b-splines approximation will produce a smooth curve through the points.

By using b-spline approximation, it can not generate a certain function to express the scattered data, but can construct an approximately curve.

B-spline curve can also be used to predict the next data from a set of known data or can be use to predict some missing data.

REFERENCE

[1] B-spline http://en.wikipedia.org/wiki/B-spline

[2] Spline http://en.wikipedia.org/wiki/Spline_(mathematics)

[3] Prautzsch, Hartmut; Boehm, Wolfgang; Paluszny, Marco. “Bezier and B-Spline Techniques”. Springer. 2002. Berlin, Germany

[4] Angel, Edward. “Computer Graphics”. Addison Wesley. 1990.

[5] Chambers, Peter; Hagen, Hans; Mclnerney, Thomas. *Introduction to Curve and Surface*. Siggraph. 1996.

[6] Farin, Gerald. “Curves and Surfaces for CAGD : A Practical Guide 5th ed”. Academic Press. 2002. San Diego, USA

[7] Farin, Gerald. *A History of Curves and Surfaces in CAGD.* Arizona State University.