2004

Bezier curves for metamodeling of simulation output

Harish J. Kingre
Louisiana State University and Agricultural and Mechanical College, hkingr1@lsu.edu

BEZIER CURVE FOR METAMODELING OF SIMULATION OUTPUT

A Thesis

Submitted to the graduate committee of the
Louisiana State University and
Agricultural and Mechanical College
In partial fulfillment of the
Requirement for the degree of
Master of Science in Industrial Engineering

In

The Department of Industrial & Manufacturing Systems Engineering

By
Harish Kingre
B.E., Nagpur University, 2001
December, 2004.

Acknowledgements

I would like to express my deep gratitude to Dr. Charles McAllister for his guidance and support throughout this work. Without his guidance and drive, I would have never reached the level of achievement or professional growth that I have attained over the last two years at LSU. Observing the critical thinking skills and clarity of thinking has improved me not only professionally, but also more importantly, personally. I would like to thank Dr. Xiaoyue Jiang for his comments and suggestions that improved the quality of this work. I would also like to thank Dr. Bhaba Sarker and Dr. James Geaghan for assisting and serving on my committee.

Finally special thanks to my parents, brothers and friends for their constant encouragement and best wishes.

Table of Contents

List of Tables

List of Figures

Abstract

Many design optimization problems rely on simulation models to obtain feasible solutions. Even with substantial improvement in the computational capability of computers, the enormous cost of computation needed for simulation makes it impractical to rely on simulation models. The use of metamodels or surrogate approximations in place of actual simulation models makes analysis realistic by reducing computational burden. There are many popular metamodeling techniques such as Polynomial Regression, Multivariate Adaptive Regression Splines, Radial Basis Functions, Kriging and Artificial Neural Networks. This research proposes a new metamodeling technique that uses Bezier curves and patches. The Bezier curve method is based on interpolation like Kriging and Radial Basis Functions. In this research the Bezier Curve method will be used for output modeling of univariate and bivariate output modeling. Results will be validated using comparison with some of the most popular metamodeling techniques.

1. Introduction

This section introduces the concept of simulation and the corresponding techniques. Next, metamodeling techniques as a tool to reduce the time and cost of computation are outlined. Last, the motivation to incorporate Bezier curves as a basis for metamodeling is discussed.

1.1 Simulation

A system can be evaluated analytically if the relationships in the model are simple enough to be represented in a mathematical form. For a real world system which is complex, obtaining an analytical solution is impossible. Such complex systems are investigated using simulation. Simulation, "is a representation of reality through the use of model or other device which will react in the same manner as reality under a given set of conditions" (Gupta and Heera, 1999).

Simulation techniques offer a number of advantages when compared with mathematical programming and standard probability analysis. The application areas for simulation are numerous. Simulation has been found to be a constructive and powerful tool in many areas. Next, some of the advantages and applications of simulation technique are discussed.

The simulation model eliminates the need for costly trial and error method of testing new concepts and configurations on physical systems and equipment. Simulation is easily understood by operating personnel and non-technical managers, which helps them to facilitate the modifications and corrections in the real systems in less time. Also, its ability to accommodate the update in real systems makes it more useful and powerful tool for designing, evaluating and analyzing. The application areas of simulation include manufacturing systems, communication systems, hardware and software systems, financial or economic systems, and new military weapon systems (Myers and Montgomery, 1995).

1.1.1 Monte Carlo Simulation Technique

Monte Carlo simulation is one of the most popular simulation techniques and has found many applications in business and industries. This technique is a very important tool in the field of operation research. Monte Carlo techniques have been used to analyze a variety of problems involving stochastic simulations and mathematical problems, which cannot be solved with analytical techniques and where physical experimentation with the actual system is infeasible. The stochastic situations are usually a long sequence of probabilistic events; though we may be able to write mathematical formulae for the probability for individual events, writing a mathematical relationship for the probability of all events in the sequential order is a complicated task. Monte Carlo is an effective technique to model a complex system involving a sequence of probabilistic events (Gupta and Heera 1999).

In spite of all these advantages simulation is not always an appropriate method for analysis, because in some situations, simulation is comparatively costlier and more time consuming for computation. To overcome these limitations, the use of metamodels has been proposed. The main objective of metamodels is to represent the relationships between the design parameters and performance measure of the system more precisely and reduce the computation burden (Kilmer and Shuman, 1997). The advantages of metamodeling techniques are evident only when simulations runs are time consuming and expensive.

1.2 Metamodeling Technique

In this section we are going to discuss about the metamodeling, the advantages of metamodeling, type of metamodeling and different criteria for selecting a metamodeling technique.

- **Metamodeling**

It is a technique in which an approximation model or metamodel or surrogate model is utilized as an alternative to the simulation model to represent the relationship between the design parameters and performance measure and to reduce the computation burden. For example, consider a given simulation model (Hussain, et al., 2002), where the input-output relationship is mathematically represented as follows:

$z = f(x)$, where x is the vector of input parameters and z is the output

The task of metamodeling is to find the way to approximate the function f that relates the input vector x to a given output z. It involves a defining function ϕ, with the predicted output response $\phi(x)$, where ϕ must approximate f with adequate precision $f(x) \cong \phi(x)$.

- **The Use of Metamodeling**

Metamodeling techniques are useful since they provide a relationship between performance measure (output) and design parameters (input). Metamodeling provide fast, reasonably priced, and computationally efficient analysis tool for optimization and design space investigation to identify the key variables. Moreover, it also integrates the discipline dependent analysis codes.

- **Type of Metamodeling Techniques**

There are number of different techniques available to fit an output model. For fast approximation of complex computer code, Response Surface Methodology (Myers and Montgomery, 1995) and Artificial Neural Network (ANN) methods are very useful and efficient in nature. The interpolation based Kriging Method (Booker at., el 1999) and Radial Basis Function approximations (Hardy, 1971) are also very popular. The statistical based Multivariate Adaptive Regression Spline (Friedman, 1991) is also very useful and efficient method. The techniques stated above are introduced in the next chapter.

- **Criteria for Selecting a Metamodeling Technique**

When selecting a metamodeling technique the following criteria may be considered ((Hussain, et al., 2002).

1. The functional form of metamodel ϕ and its implementation complexity.
2. The set of input points used to determine the coefficient of the metamodels.
3. Assessment of the adequacy of the fitted model quantified through different measures, such as lack of fit, cross validation, and other model diagnostics.
4. Ability to gain insight into the behavior of the simulation models using the fitted metamodel.
5. Ability to capture the shape of arbitrary smooth functions based on observed values, which may be perturbed randomly.
6. Robustness of the prediction away from observed points.
7. Computational stability in predicting the fitted metamodels due to small changes in parameters defining the metamodel ϕ .
8. Existence of software for computing the metamodels and characterizing its fit and prediction error.

1.3 Motivation and Goal of Research

We are employing Bezier curves to introduce a new metamodeling technique. Prior application of Bezier curves in the field of input modeling and additional properties such as interpolation, convex hull, and local control are motivating factors to develop Bezier curves for implementation in a new metamodeling technique. The results obtained from a test case using data collected from a simulation output of a manufacturing process indicate that Bezier fit for univariate output modeling is comparatively better than the polynomial regression technique. The final goal of this research is to development of a metamodeling technique for univariate and bivariate model based on the principal of Bezier curve and to validate the outcome by comparing it with some of the popular metamodeling technique like radial basis function (RBF) and polynomial (PR) metamodels.

2. Metamodeling Techniques

This section introduces a number of different metamodeling techniques with corresponding mathematics. In engineering design, metamodeling techniques have been widely used for optimization of systems and to reduce solution time of analysis that involves computationally expensive and time consuming simulation programs (Barton, 1998). A comprehensive review of metamodeling applications in mechanical and aerospace has been provided in Simpson, et al (1997), and a comparative study of various metamodeling techniques has been provided by Jin, et al. (2000) using different modeling criteria on a variety of test problems.

2.1 Metamodeling Techniques for Simulation

The metamodels can be classified into parametric and non-parametric techniques (Hussain, et al., 2002). Parametric techniques approximate functions a-priori without any prior knowledge about the underlying data. These functions are used to fit the observed response by adjusting the coefficient of chosen functions. Examples are polynomial models, general linear models, non-linear models, and Taguchi models. Non-parametric techniques do not have an a-priori set of functions; instead they use an a-priori method for constructing an approximating function based on observed responses. Examples are radial basis function, and spline based models.

This section presents metamodeling techniques for simulations, such as Response Surface Methodologies, Kriging, Multivariate Adaptive Regression Spines, Radial Basis Functions and Artificial Neural Networks.

2.1.1 Response Surface Metamodeling

Response surface metamodeling is a collection of statistical and mathematical modeling techniques and is the most popular metamodeling technique. It has been used effectively for constructing simple and fast approximations of complex computer codes (Mahidi, et al., 1999) over the last thirty years.

The model is of the form:

$$y(x) = f(x) + \varepsilon \text{, where } y \text{ is a scalar and } \varepsilon \text{ is a scalar} \qquad (2.1)$$

This approach generally fits better using first or second order polynomials of *y* (output). In the case of higher order polynomials unsteadiness may arise (Barton, 1992), or it may be too complicated to sample adequate data to approximate all of the coefficients in the polynomial equation.

- **Response Surface Method (RSM)**

Initially the goal of response surface modeling was to analyze the outcomes of physical experiments and generate empirically-based models of the observed outcomes (response values). RSM is represented by Equation (2.1) where *y(x)* is the unidentified function of interest, *f(x)* is a known polynomial function of *x*, and ε is a random error which is assumed to be normally distributed with mean zero and variance σ^2. The individual error ε_n at each observation is also assumed to be independent and identically distributed. The polynomial function *f (x)* is a low order polynomial, and it is used to approximate *y(x)*, which is assumed to be either linear, Equation (2.2) or quadratic, Equation (2.3) (Myers and Montgomery, 2000).

$$\hat{y} = \beta_0 + \sum_{n=1}^{k} \beta_n x_n \qquad (2.2)$$

$$\hat{y} = \beta_0 + \sum_{n=1}^{k} \beta_n x_m + \sum_{n=1}^{k} \beta_{nn} x_n^2 + \sum_{n} \sum_{m} \beta_{nm} x_n x_m \qquad (2.3)$$

The least square regression (LSR) method is used to determine the coefficients $\beta_0, \beta_n, \beta_{nn}$ and β_{nm}. The LSR method minimizes the sum of squared deviation of predicted values $\hat{y}(x)$ from the actual values *y(x)*. The β coefficients can be found using least square regression given by

$$\beta = \left[X^{'} X\right]^{-1} X^{'} y \tag{2.4}$$

In Equation (2.4), *X* is the input matrix of test data points and *y* is a column vector containing the performance measure at each test point (Mahidi, et al., 1999 and Myers, et al., 2002).

2.1.2 Kriging Metamodeling

Kriging is also called Design and Analysis of Computer Experiment (DACE) and is based on interpolation (Simpson, et al., 1998). Kriging's model validity is independent of random error. It may be better suited for applications involving computer experiments because it can either "honor the data" or provide an exact interpolation or "smooth the data" (Simpson, et al., 1998 and Cressie, et al., 1993).

The use of Kriging is restricted to a very small region because it involves inversion and multiplication of numerous matrices. Additionally it does not have a closed-form representation and is unavailable in commercial software. Comparison of Kriging with polynomial regression has been provided in (Simpson, et al., 1998).

- **Kriging Method**

The model is of the form equation given below; it postulates a combination of global function (*f(x)*) and departure (*Z(x)*) (Jin, et al., 2000):

$$y(x) = f(x) + Z(x) \tag{2.5}$$

In Equation (2.5) *y(x)* is an unknown function of interest, *f(x)* is a known and is generally a polynomial function of *x*, and *Z(x)* is assumed to be a realization of a stochastic process with mean zero, variance σ^2 and having spatial correlation function (Jin, et al., 2000) on covariance matrix of *Z(x)* is given by

$$\text{Cov}\,[Z\,(x_n),\, Z\,(x_m)] = \sigma^2\,\mathbf{R}[R\,(x_n,\, x_m)] \tag{2.6}$$

In Equation (2.6), **R** is a correlation matrix and R ($x_{n,}$ x_m) is a correlation function. A range of correlation functions can be selected (Simpson, et al., 1998). The ability to use a wide range of correlation functions makes the method extremely flexible. Despite many advantages such as a stepwise algorithm to select the vital factors and screening input factors to build a predictor model (Jin, et al., 2000), applications of Kriging are limited because of the time consuming computation of maximum likelihood (θ) which is used to fit the model in the *K*-dimensional optimization problem. A set of problems solved using Kriging method is provided by (Jin, et al., 2000).

2.1.3 Multivariate Adaptive Regression Spline Metamodeling

Multivariate Adaptive Regression Spline (MARS) metamodeling is a novel statistic method presented in (Friedman, 1991). MARS attempts to approximate complex relationships by a series of linear regressions on different intervals of the independent variable ranges or subregions of the independent variable space. It is very flexible as it can adapt to any functional form. MARS builds a relation from a group of coefficients (basis functions) that are entirely determined from the regression data. For approximation MARS uses a two sided truncated function also called a forward-backward iterative approach.

MARS partitions the input space into ranges or subregions, each with its individual regression equation. This makes MARS mostly appropriate for problems with higher input dimensions. It also provides better results in the case of sparse input (Jin, et al., 2000).

- **MARS Method**

A MARS model can be postulated as:

$$\hat{y} = \sum_{n=1}^{N} a_n B_{kn}(x_{v(k,n)}) \qquad (2.7)$$

In Equation (2.7) $\hat{y}$ is the dependent (outcome) variable, a_n is the coefficient of expansion, and B_{kn} is the basis function, defined as:

$$B_{kn}(x_{v(k,n)}) = \prod_{k=1}^{K} h_{kn}, \qquad (2.8)$$

where x is the predictor variable, k is the order of the interaction and n represents number of terms. For the first order *(k=1)* of interaction, the model is additive, and for greater than one *(k>1)*, the model is pair-wise interactive (Friedman, 1991) MARS searches the entire design space, and during this search of an increasing large number of basis functions are added to the model. MARS automatically determines the most important independent variables as well as the most significant interaction between the independent and dependent variables.

MARS is relatively new compared to the other techniques. The accuracy and reduction of computation cost are the prime advantages of using MARS.

2.1.4 Radial Basis Function (RBF) Metamodeling

The Radial Basis Function metamodeling technique is based on interpolation. It provides another approach to multivariate metamodeling. In an experimental comparison (Franke, et al., 1982), it is found that the RBF is better than the Response Surface Methodology, Multivariate Adaptive Regression Spline, and Kriging. Since it is an interpolation method its direct approach to simulation metamodeling is also restricted to a small region like Kriging.

- **RBF Method**

The RBF model is constructed as a linear combination of K Radial Basis Functions with K centers. Radial basis functions based on the Euclidean distance a metric to approximate output (performance measure). The original development by Hardy (1971) introduced, among others, simple "multiquadratic" basis functions:

$$\hat{y} = \sum_{j=1}^{K} \beta_j \left\| x - x_{ej} \right\| , \qquad (2.9)$$

where β_j is the weight or coefficient associated to Radial Basis Functions that is centered at x_{ej}, and x_{ej} is the experimental input. The $\hat{y}$ and β_j in the metamodel, depend on location of the observed input x_{ej}. The coefficients β_j are found by replacing left hand side of (2.9) with $f(x_j), i = 1,...,n,$ and solving the resulting linear system.

The "multiquardratic" basis function provides a good result for bivariate and higher order models. Radial basis functions are also related to another class of spline functions. The so called "thin plate splines" have radial basis functions of $\left\| x - x_j \right\|^2 \log \left\| x - x_j \right\|$; it also provides a better fit to bivariate models with scattered input. Despite a lack of direct application to simulation modeling RBF provides the best solutions for small and sparse sample test problems. In (Jin, et al., 2000) it is concluded that RBF performs best when both average accuracy and robustness are considered.

2.1.5 Artificial Neural Network (ANN) Metamodeling

Like Response Surface Methodology the Artificial Neural Network (Mahidi, et al., 1999) is also a widely applied metamodeling technique to generate approximations of complex computer codes. It eliminates the possibility of pre-selecting an incorrect functional form, due to its ability to universally model any relationship through a neural network with non linear transfer function. This means the error component of Equation

(2.1) would be theoretically zero. The other advantages of ANN include. (*i*) ANN is not responsive to deviations from traditional statistic model hypothesis, (*ii*) Gaussian distribution of error, and (*iii*) ANN can model a combination of continuous and discrete numerical variables. Moreover, most ANN paradigms are global models (Mahidi, et al., 1999); thus a particular neural network can be developed to model the complete simulation response surface, and ANN can also be viewed as flexible computing parallel devices for producing performance measures that are complex functions of multivariate input information.

Similar to the other models ANN is also dependable on the observed input. The accuracy and precision of ANN is based on the quality of the data set used in modeling. Similar to high-order polynomial regression model the precision and accuracy of the model depreciates when small data sets are used for modeling.

3. Bezier Curves and Bezier Surfaces

In 1970, Pierre Bezier at Renault and Paul De Casteljau at Citroen independently developed Bezier curves for CAD/CAM operations. A simple Bezier Curve is represented by:

$$P(t) = \sum_{i=0}^{n} P_i B_{i,n}(t), \tag{3.1}$$

The parametrically defined polynomial, *P (t),* is a class of approximating splines.

$t \in (0,1)$ and P_i is the number of control points for $i = 0,\ldots,n.$

The Blending function, $B_{i,n}(t)$, is the Bernstein polynomial:

$$B_{i,n}(t) = \frac{n!}{i!(n-1)!} t^i (1-t)^{n-i}, \tag{3.2}$$

In general there are *n+1* Bernstein polynomials of degree *n*. For example for *n=2*

$$B_{0,2}(t) = (1-t)^2, B_{1,2}(t) = 2t(1-t), B_{2,2}(t) = t^2, \tag{3.3}$$

3.1 Properties and Characteristics Bezier Curve.

This section presents properties of Bernstein polynomials and Bezier curves. Next, the composite Bezier curve is introduced. Last, the properties of Bezier Surface and applications of Bezier Curves are described.

3.1.1 Properties of Blending Function or Bernstein Polynomial

In this section we are going to introduced five very important Properties of Bernstein polynomials.

- **Recurrence Relation**

Bernstein polynomial can be generated in the following way.

Set $B_{0,0}(t) = 1$ and $B_{i,n}(t) = 0$ for *i<0* or *i >n*, and use of recurrence relation

$$B_{i,n}(t) = (1-t)B_{i,n-1}(t) + tB_{i-1,n-1}(t), \tag{3.4}$$

For $i = 1, 2 \ldots, n-1$.

- **Nonnegative on [0,1]**

Bernstein polynomials are nonnegative over the interval [0, 1] see Figure 1

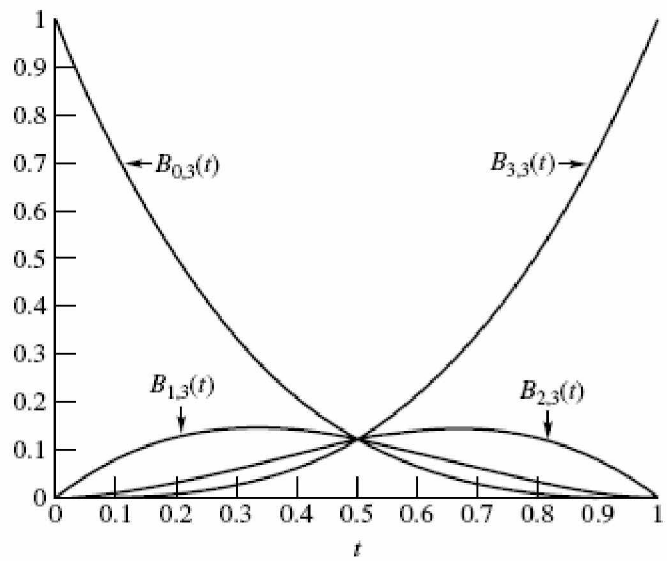


Figure 1: Variation of Bernstein Polynomial function

- **The Bernstein polynomials form a partition of unity**

$$\sum_{i=0}^{n} B_i, n(t) = 1, \tag{3.5}$$

Substitute $x=t$ and $y= 1-t$ in to the binomial theorem

$$(x+y)^n = \frac{n!}{i!(n-i)!} x^i y^{(n-i)} = [t+(1-t)]^n = 1^n = 1, \tag{3.6}$$

- **Derivatives of Bernstein Polynomial Function**

$$\frac{d}{dt} B_{i,n}(t) = n(B_{i-1,n-1}(t) - B_{i,n-1}(t)), \tag{3.7}$$

The above property is very useful to determine the continuity of the Bezier curve and it also helps to fit a PDF for a Bezier distribution.

- **Basis**

The Bernstein polynomial of order n $(B_{i,n}(t)$ for $i = 0,1,\ldots,n)$ forms a space basis for all polynomials of degree less than or equal to *n*. This property states that any polynomial of degree less then or equal to n can be written uniquely as a linear combination of the Bernstein polynomials of order *n*.

3.1.2 Property of Bezier Curve

In this section we are going to introduced four very important properties about Bezier Curve.

- **Affine invariance**

The affine invariance property indicates that Bezier curves are invariant under affine maps (Gerald Farin, 2002). This means that the next two procedures will yield the same result,

1) First compute all control points and then apply an affine map to it.

2) First, apply an affine map to the control polygon and then evaluate the mapped polygon at parameter values.

The significance of an affine invariance can be evaluated with the help of the following example. Suppose we plot a cubic curve b^3 by evaluating it at 50 control points and then plotting the resulting array of all 50 points. We would like to plot the curve with rotation, and there are two ways to do this:

1) Rotate each control point, and then plot.

2) Rotate four control points on the curve. Evaluate it 50 times and then plot.

In first case it needs 50 iterations to complete rotation, whereas in the second case only $4!$ are needed.

- **Invariance under affine parameter transformation**

One may think that it is necessary to define Bezier curves over interval [0, 1], however this is not mandatory; it can be defined over any arbitrary interval [a, b]. Therefore, $a \le u \le b$ and $t = (u\text{-}a)/\,(b\text{-}a)$

The transition from the interval [0, 1] to the interval [a, b] is an affine map; therefore, we can say that the Bezier curves are invariant under affine parameter transmission. For more details see (Farin Gerald, 2002).

- **Convex Hull property**

This property indicates that the plot generated by a Bezier curve of degree *n* is a continuous curve, bounded by the set of control points, and the curve will begin and end at the points P_0 and P_n, respectively.

The practical aspects of convex hull are:

1) The shape of curve can be modified by making a small adjustment to the control points.

2) Interference checking,

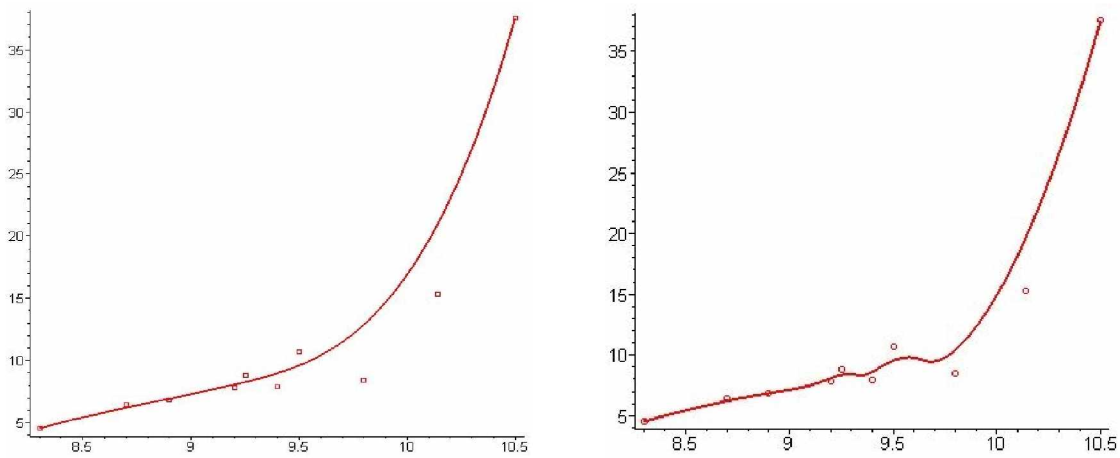
This property plays an important role, suppose Bezier curves are selected to determine the moment of a Robot arms, to avoid the collision of Robot arms we can circumscribe the smallest possible box around the control polygon of each curve such that its edges parallel to some coordinate system.

- **Endpoint Interpolation**

The Bezier curve passes through P_0 and P_n: it implies that the end points of a curve are certainly two critical points. It is therefore necessary to have direct control on these points, which is assumed by endpoint interpolation.

3.1.3 Composite Bezier Curve

Because the Bezier curve can be define over any arbitrary interval [a, b], a curve defined over interval [0, 1] can be subdivided into many regions. Therefore in order to generate shapes that are too complex for a single Bezier curve, a composite Bezier curve can be generated by a union of several Bezier curves. When using a composite Bezier, adjacent curves are joined by common point that is shared by both curves. If we want to ensure that two pieces meet smoothly, the first, last, and common point must be collinear. The difference between a simple Bezier curve (Figure 2a) and a composed Bezier curve (Figure 2b) with same control point are easily visible. The proposed research, the output model will be fit using both a simple Bezier curve and a Piecewise Bezier curve.



a. Simple Bezier curve b. Piecewise Bezier curve

Figure 2: Bezier Curves

3.2 Bezier Surface

Bezier Surfaces or patches form an extension of Bezier curves. The primary difference when implementing Bezier patch compare to Bezier curves is use of a control points array and a Bivariate Bernstein polynomial. The edges of Bezier surfaces are

Bezier curves and the corner control points are always on the surface. The Bezier surfaces are represented by:

$$P(t_x,t_y)=\sum_{i=0}^{n_x}\sum_{j=0}^{n_y}B_{n_x,i}(t_x)B_{n_y,j}(t_y)P_{ij} \quad , \tag{3.8}$$

where t_x and $t_y \in (0,1)$, and the other parameters are same as previously defined.

Similar to Bezier curves the first and the last points lie on the edge of a Bezier surface. A simple Bezier surface is shown in Figure 3. This surface is controlled by 16 control points ($P_{0,0}$ to $P_{3,3}$) and it is enclosed by four Bezier curves.

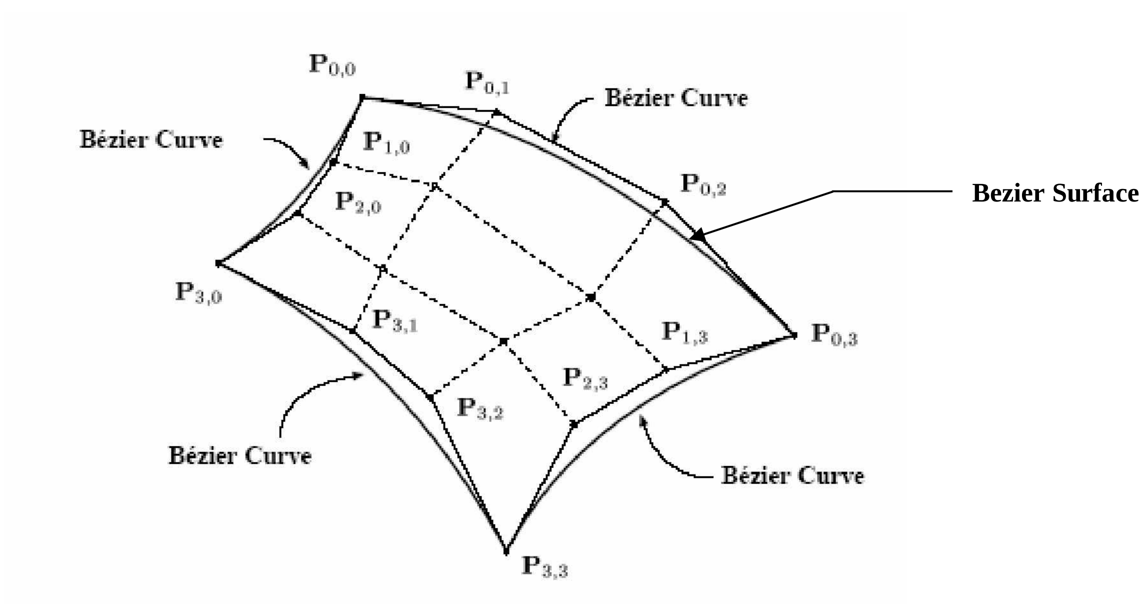


Figure 3: Bezier Surface as an extension of Bezier curve

3.2.1 Properties of Bezier Surfaces

Like Bezier curves the Bezier patches have the same properties of affine invariance and convex hull, which are already declared in Section 2.2.2. In this section we are going to outline the new property of Bezier patches namely boundary curve

- **Boundary Property**

The boundary curves of the Bezier patch are Bezier curves. The control polygons are specified by the boundary polygons of the control net. In particular, the four corners of the control net all lie on the same patch (Gerald Farin, 2002).

3.2.2 Further Investigation in Bezier Surface

In this section the two types of Bezier surfaces are introduced. The two main types of Bezier surfaces use rectangular and triangular Bezier patches.

1. The Rectangular Bezier Patch

The rectangular Bezier Patch is found as a univariate Bernstein polynomial function with P_{ij} control points and $n \times m$ degrees:

$$P(s,t) = \sum_{i=0}^{n} \sum_{j=0}^{m} R_{ij} B_i(s) B_j(t), \text{ where } 0 \le s \le 1 \text{ and } 0 \le t \le 1 \qquad (3.9)$$

2. The Triangular Bezier Patch

The Triangular Bezier patch is found as a bivariate Bernstein polynomial function of degree n with $T_{i,j,K}$ control points:

$$P(s,t,u) = \sum_{i+j+,k=n} T_{i,j,k} B_{i,j,k}(s,t,u), \text{ where } s,t,u \ge 0 \text{ and } s+t+u=1 \quad (3.10)$$

Due to different geometry in both in the rectangular and triangular approaches, it is difficult to use both in a single Bezier fit.

3.3 Application of Bezier Curves

This section outlines applications of Bezier curve in the area of computer graphics and input modeling. Then, the univariate Bezier distribution and bivariate Bezier distributions are explained.

Computer Graphics: Bezier curves are utilized extensively in computer graphics, due to their simplicity of implementation, insightful construction, and numerical steadiness.

Input Modeling: There are number of procedures for modeling a simulation input process. These procedures include informal graphical techniques based on probability plots and statistical goodness of fit test such as the Kolmogorov-Smirnov, Chi-square, and Anderson-Darling test, statistics unfortunately none of these procedures is certain to yield a definitive conclusion (Wagner and Wilson 1996). In 1996 Wagner and Wilson developed a flexible, interactive, graphical methodology for modeling a broad range of inputs processes. They exploited the properties of Bezier curves to develop a distribution family that has an open ended parameterization capable of generating an unlimited number of distribution shape (Wagner and Wilson 1996).

Bezier curves are often used to approximate a smooth univariate function on a restricted interval by requiring the Bezier curve to pass in the proximity of selected control points.

3.3.1 Univariate Bezier Distributions

In the univariate Bezier distribution, the distribution function is given by (3.10) and corresponding probability distribution function is:

$$P'(t) = \frac{\sum_{i=0}^{n-1} B_{n-1,i}(t)\Delta z_i}{\sum_{i=0}^{n-1} B_{n-1,i}(t)\Delta x_i}, \tag{3.11}$$

where $\Delta x \equiv (\Delta x_0,\ldots,\Delta x_{n-1})^T$ and $\Delta z \equiv (\Delta z_0,\ldots,\Delta z_{n-1})^T$

Justification of Equation (3.11) is elaborated in Farin (1990).

According to (Wilson and Wagner, 1996) this distribution family has a natural, extensible parameterization that allows unlimited flexibility in representing probabilistic behavior of real world processes.

3.3.2 Bivariate Bezier Distributions

The bivariate Bezier distribution function for *x* and *y* is given by:

$$P(t_x) = \sum_{i=0}^{n_x} B_{n_x,i}(t_x)x_i \ \ , \forall \ t_x \in [0.1], \tag{3.12}$$

$$P(t_y) = \sum_{i=0}^{n_x} B_{n_x,i}(t_y)y_i \ , \forall t_y \in [0,1], \tag{3.13}$$

The actual distribution function in *x* and *y* is represented by:

$$P(t_x, t_y) = \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} B_{n_x,i}(t_x), B_{n_y,j}(t_y)\, p_{i,j} \quad , \quad \forall t_x, t_y \in [0,1] \tag{3.14}$$

Corresponding probability function is given by:

$$P'_{x,y}(t_x, t_y) = \frac{\sum_{i=0}^{n_x-1} \sum_{j=0}^{n_y-1} B_{n_x-1,i}(t_x) B_{n_x-1,i}(t_x) \Delta_i \Delta_j\, p_{i,j}}{\sum_{i=0}^{n_x-1} B_{n_x-1,i}(t_x)\Delta x_i \sum_{j=0}^{n_y-1} B_{n_y-1,j}(t_y)\Delta y_j}, \forall t_x t_y \in [0,1] \tag{3.15}$$

According to Wagner and Wilson (1996), the method presented in their paper provides a representation that is well suited to graphical interactive simulation input modeling.

3.4 Summary

Advantages of metamodeling techniques are characterized by significant reduction in time and cost of computation. These Metamodeling techniques are based on the principle of interpolation, regression, and artificial neural networks. Design strategies for metamodeling techniques include some of these principles. The Bezier Curves are also based on the principle of interpolation. Applications of Bezier curves in input modeling motivated the introduction of Bezier Curve for developing a new metamodeling technique.

4 Research and Development of Bezier in the Field of Metamodel

The proposed research seeks to develop a new metamodeling technique for univariate and bivariate model based on the Bezier Curve. A variety of metamodeling techniques presently exist, and each technique is based on different concepts such as; regression, neural network, interpolation, and statistics.

The research develops a new metamodeling technique for univariate and bivariate output modeling. The Bezier Curve is employed for univariate output modeling and Bezier patches are employed for bivariate output modeling. The main goal of this research is to propose a new metamodeling technique for univariate and bivariate output modeling that is less time consuming than direct computation. For example, Task 1 seeks to implement the Bezier curve for metamodeling. Validation of Bezier fit will be done via comparison with some of the most popular existing techniques. The results of each modeling techniques will be evaluated with respect to accuracy and robustness. Further development to improve the test result via a piecewise Bezier curve is discussed, which is the composite Bezier Curve addressed in Section 3. Next, a bivariate output model (Figure.5) incorporating Bezier Patches or Bezier surfaces is discussed. Last, the results obtained for the univariate output and bivariate output modeling via Bezier curve and Bezier surfaces are elaborated in the final section. Figures 4 and 5 illustrates basic frameworks for developing a metamodel for univariate and bivariate output model using Bezier curves and patches.

For Univariate Output Modeling

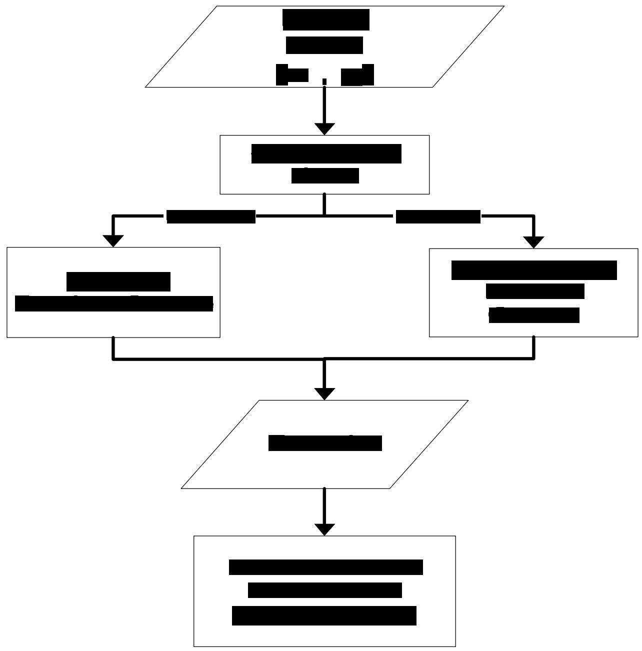


Figure 4: Task chart for univariate output modeling

For Bivariate Output Modeling

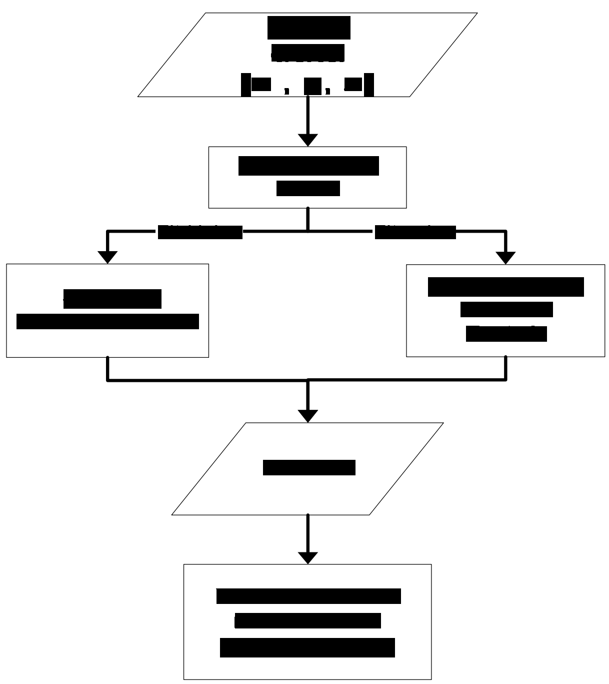


Figure 5: Task chart for bivariate output modeling

4.1. Research Tasks

This section presents detailed research tasks that are completed in order to realize the proposed metamodeling techniques.

4.1.1 Investigation of Bezier Curve for Metamodeling

Sections 1 and 2 suggest the availability of a number of different methods for metamodeling. The method explained in these sections possesses some properties that made those methods attractive for metamodeling. Similarly, some properties as well as outcomes obtained in investigation Bezier curve illustrate its compatibility for metamodeling.

4.1.2 Univariate Output Model Fitting Using Bezier Curve

The applications of Bezier Curve in input modeling and its properties such as interpolation, convex hull and global control motivated us to utilize Bezier Curve for univariate output modeling. A simple Bezier curve is used for univariate output modeling. Parameters of Bezier curves are Significant and can be easily estimated through the data points. Additionally, Bezier Curve has an open-ended parameterization.

As discussed in the third section the Bezier Curve usually interpolates in [0, 1] interval therefore, to ensure that *P (t)* is a continuously increasing function, inputs must follows the form $x_0 \leq x_1 \leq x_2 .... \leq x_n$ (Wagner & Wilson 1996). Thus, the first step will be to sort the output corresponding to ascending order, and then the simple Bezier curve is fit. The Bezier curve employed to fit the univariate output model is represented by:

$$P(t) = \sum_{i=0}^{n} P_i B_{i,n}(t)$$

Initially, we used a simple Bezier fit for the univariate output modeling. In the simple Bezier fit there is no subdivision of the entire curve. The procedures for fitting a simple and composite Bezier curve and patches to a univariate or bivariate output model are described in the next section.

4.1.3 Procedure for Univariate Output Modeling Using Bezier Curve

MAPLE 8.0 will be used to develop code for fitting Bezier Curves and Bezier patches to univariate and bivariate output models respectively. As explained in the previous section, the first step will be to sort input into ascending order, which is followed by entering the test points in the MAPLE code to construct the plot for all the control points.

Further investigation in Bezier Curve shows that, a composite Bezier curve gives a better solution than a simple Bezier curve for a problem with one input and one output. In next section a procedure to fit a Composite Bezier curve is explained.

4.1.4 Procedure for Univariate Output Modeling Using Composite Bezier Curve

Composite Bezier curves follows the same procedure explained in the previous section. The primary difference being in composite Bezier curve compare to simple Bezier curve is subdivision of curve based on the number of control points and degree of Bezier curve. This research uses a cubic Bezier Curve in piecewise fitting of a Bezier curve to univariate output model. The concept of composite Bezier curve is outlined in section 3.

4.1.5 Bivariate Output Modeling Using Bezier Patches

Bivariate output modeling is similar to the univariate output modeling. The only difference between the bivariate and univariate output modeling is the number of Design parameters**.** The Bezier patch employed to fit the bivariate output model is represented by:

$$P(t_x, t_y) = \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} B_{n_x,i}(t_x) B_{n_y,j}(t_y) P_{ij}$$

4.1.6 Procedure for Bivariate Output Modeling Using Bezier Patch

MAPLE 8.0 will be used to develop code for fitting Bezier Curves and Bezier patches to bivariate output models respectively. As explained in the previous section, the first step will be to sort input into ascending order, which is followed by entering the test points in the MAPLE code to construct the plot for all the control points.

4.2 Validation Strategies

This section addresses validation of new Bezier curve metamodeling technique via comparative investigation. For the comparative investigation multiple metamodeling method will be used. The outcome of each metamodeling technique is evaluated using the following aspects.

4.2.1 Accuracy

Accuracy is "the quality or nearness to the true value." In terms of modeling, "it is a capability of predicting the performance measure of the system over the design space of interest" (Jin, et al., 2000).

To access the accuracy of newly predicted points three different metrics are used: R square, relative average absolute error, and relative maximum absolute error, which are described next.

- **R square:** It is also referred to as proportion of variance explained by the model, the equation for this measure is:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{SSE}{SST} \ , \tag{4.1}$$

where $\hat{y}$ is the corresponding predicted value for the observed value y_i and $\bar{y}$ is the mean observed value. While SSE (sum of squared error) represents the departure of the metamodel from the real simulation model, the SST (total sum of squared error) captures how irregular the problem is. The larger the value of R square, the more accurate the metamodel.

- **Relative Average Absolute Error (RAAE):** The equation for this measure is:

$$RAAE = \frac{\sum_{i=1}^{n}\left|y_i - \hat{y}\right|}{n * SD}, \tag{4.2}$$

Where SD stands for standard deviation. The RAAE is highly correlated with the R square. The smaller the value of RAAE, the more accurate the metamodels.

- **Relative Maximum Absolute Error (RMAE):** The equation for this measure is:

$$RMAE = \frac{\max\left(\left|y_{1-}\hat{y}_1\right|, \left|y_{2-}\hat{y}_2\right|, \ldots, \left|y_{n-}\hat{y}_n\right|\right)}{STD}, \tag{4.3}$$

The large value of RMAE indicates a large error in one region of the design space. Therefore a small RMAE is preferred.

4.2.2 Sum of Squared Error

The sum of squared error gives the deviation of metamodel from actual simulation model. The equation for SSE is represented by:

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}), \qquad (4.4)$$

The smaller the value of SSE, the more accurate the metamodels.

5 Results and Validation

In this section the results of a univariate and bivariate output model using Bezier fit are presented. Encouraging results obtained by the initial investigation motivated development of higher dimension output modeling using Bezier curves.

5.1 Univariate Output Model

Input Data points for the Bezier curve is given in Table 1. These points are taken form a Manufacturing simulation problem.

Table 1: Actual Data Points

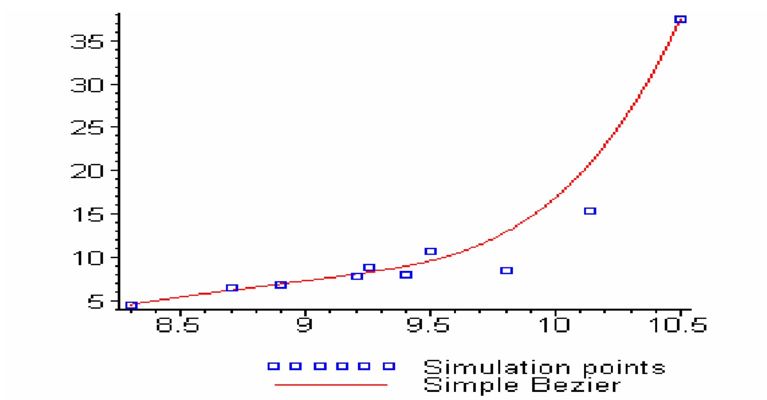
X	Y
8.3	**4.5471**
8.7	**6.4568**
8.9	**6.7928**
9.2	**7.8063**
9.25	**8.784**
9.4	**7.911**
9.5	**10.661**
9.8	**8.4412**
10.14	**15.341**
10.5	**37. 547**

5.1.1 Simple Bezier Fit to Univariate Output Model

We employed a Maple 8.0 to fit a Bezier curve to the univariate output model. The results of univariate output modeling using Bezier curve are given in Table 2.

Table 2: Bezier fit

X	Y
8.3	**4.5471**
8.7	**6.2056**
8.9	**6.94**
9.2	**8.064**



9.25	**8.273**
9.4	**8.9778**
9.5	**9.649**
9.8	**12.945**
10.14	**21.0115**

Figure 6: Actual data Pints

Figure 6 shows the simple Bezier curve for univariate output model, the points in Figure 6 are the actual Data points.

5.1.2 Piecewise or Composite Bezier Fit to Univariate Output Model

The Table 3 shows the outcome of a univariate output model using a composite Bezier curve.

Table 3 composite Bezier fit

X	Y
10.14	**4.5471**
9.5	**6.24**
8.3	**6.78**
8.7	**8.03**
9.2	**8.38**
9.8	**8.5**
10.5	**9.63**
9.25	**10.34**
8.9	**18.94**
9.4	**37.547**

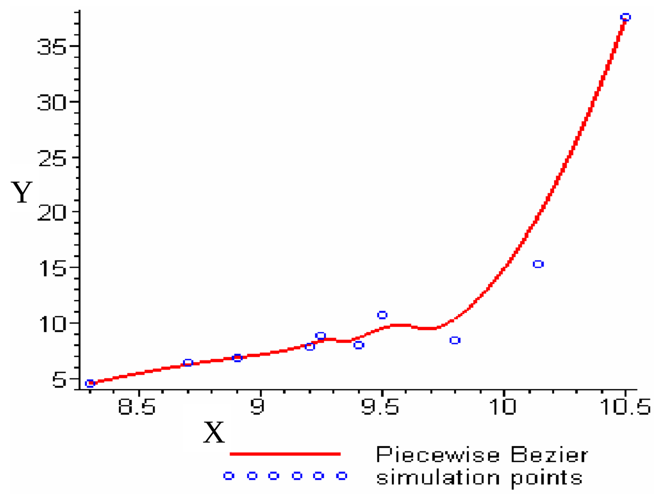


Figure 7: A composite Bezier Curve

Figure 7 shows the composite or piecewise Bezier curve for univariate output model. The points in Figure 7 are the actual Data points.

5.1.3 Validation of Univariate Model

The outcomes of the simple Bezier fit and composite Bezier fit is validated via comparison with other metamodeling technique. The performance of Bezier fit is measured via accuracy. To access the performance of Bezier curve, three different metrics are used: R square, relative average absolute error, and relative maximum absolute error. The robustness of Bezier curve is also measured by means of comparison of sum of squared error of Bezier curve and Polynomial regression. The results obtained from the univariate output model using polynomial regression metamodeling technique are presented in Table 4.

Table 4: Polynomial Regression

X	Y
8.3	**7.142**
8.7	**12.073**
8.9	**12.355**
9.2	**11.72**
9.25	**11.61**
9.4	**11.45**
9.5	**11.6**
9.8	**14**
10.14	**22.31**
10.5	**40.7**

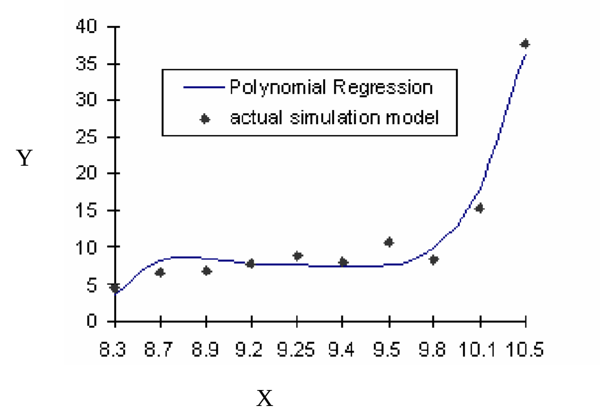


Figure 8: A polynomial regression fit

In Figure 8 the outcome of the polynomial regression is plotted with the actual data points.

5.1.4 Performance of Bezier Fit for Univariate Model

R square, relative average absolute error and relative maximum absolute error of simple Bezier fit and piecewise Bezier fit is compared with polynomial regression in Table 5.

Table 5: Performance of Bezier fit

Accuracy	**Bezier Fit**		**Polynomial Regression**
	Simple	**Piecewise**	
R square	0.903	0.968	0.948
RAAE	0.139	0.102	0.422
RMAE	0.589	0.508	0.724

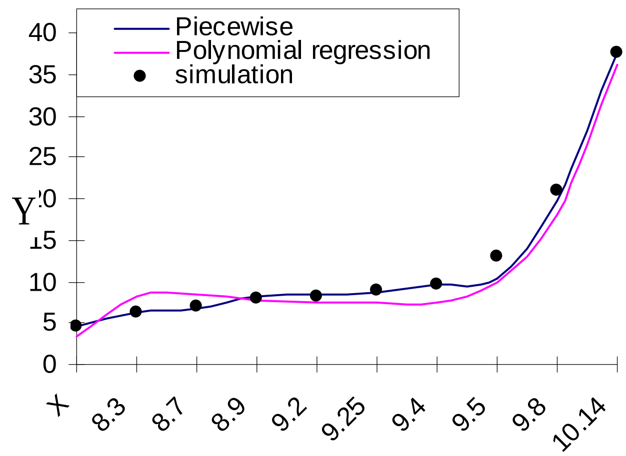


Figure 9: comparison of Polynomial Regression and Bezier Curve.

As shown in Table 5,

R square $_{(\textbf{Composite})}$ > R square $_{(\textbf{Polynomial Regression})}$ >R square $_{(\textbf{simple})}$,

RAAE $_{(\textbf{Composite})}$ < RAAE $_{(\textbf{Simple})}$ < RAAE $_{(\textbf{Polynomial Regression})}$, and

RMAE $_{(\textbf{Composite})}$ < RMAE $_{(\textbf{Simple})}$ <RMAE $_{(\textbf{Polynomial Regression})}$.

The results in Table 5 and Figure 9 clearly depict that the piecewise Bezier fit is better than the polynomial regression.

5.1.5 Robustness of Bezier Fit for Univariate Model

To measure the robustness, six new test points from the actual simulation is used. This new six points are given in Table 6.

Table 6: Test points

X	Y
8.5	**6.01**
8.8	**6.87**
9.1	**6.43**
9.6	**7.24**
9.9	**10.11**
10.2	**24.89**

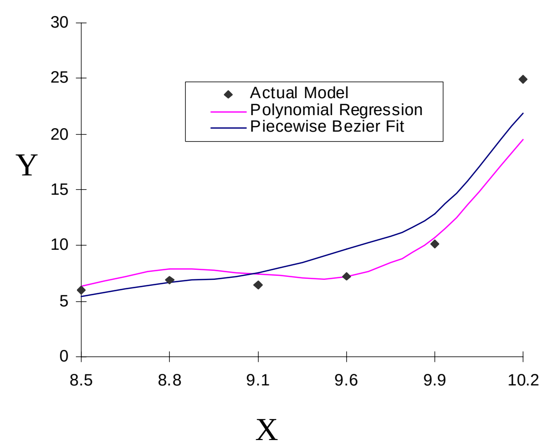


Figure 10: Prediction of test points with different metamodeling techniques.

To measure the robustness of the Bezier Fit, the SSE of polynomial regression is compared with simple and Piecewise Bezier, which is shown in Table 7, 8 and 9.

Table 7: SSE of polynomial Regression

X	Y	$\hat{Y}$	Residual	$(\text{Residual})^2$
8.5	**6.01**	**6.302875**	**-0.29287**	**0.085776**
8.8	**6.87**	**7.946752**	**-1.07675**	**1.159395**
9.1	**6.43**	**7.431061**	**-1.00106**	**1.002123**
9.6	**7.24**	**7.224576**	**0.015424**	**0.000238**
9.9	**10.11**	**10.76371**	**-0.65371**	**0.427335**
10.2	**24.89**	**19.50353**	**5.386472**	**29.01408**
Sum of Squared error				**31.68895**

Table 8: SSE of Simple Bezier

X	Y	$\hat{Y}$	Residual	$(\text{Residual})^2$
8.5	**6.01**	**5.58**	**0.43**	**0.1849**
8.8	**6.87**	**6.72**	**0.15**	**0.0225**
9.1	**6.43**	**7.63**	**-1.2**	**1.44**
9.6	**7.24**	**10.36**	**-3.12**	**9.7344**
9.9	**10.11**	**14.45**	**-4.34**	**18.8356**
10.2	**24.89**	**22.87**	**2.02**	**4.0804**
Sum of Squared error				**34.297**

Table 9: SSE of Composite Bezier

X	Y	$\hat{Y}$	Residual	$(\text{Residual})^2$
8.5	**6.01**	**5.503**	**0.507**	**0.257049**
8.8	**6.87**	**6.624**	**0.246**	**0.060516**
9.1	**6.43**	**7.503**	**-1.073**	**1.151329**
9.6	**7.24**	**9.751**	**-2.511**	**6.305121**
9.9	**10.11**	**12.288**	**-2.178**	**4.743684**
10.2	**24.89**	**22.823**	**2.067**	**4.272489**
Sum of Squared error				**16.79019**

As shown in Tables 7, 8, and 9:

$$\text{Residual}^\wedge 2_{(\textbf{Composite})} < \text{Residual}^\wedge 2_{(\textbf{Polynomial Regression})} < \text{Residual}^\wedge 2_{(\textbf{Simple})}$$

Tables 7-9 and Figure 10 clearly show that Piecewise Bezier is comparatively more robust than simple Bezier curve and polynomial regression.

5.2 Bivariate Model

For bivariate output model four different functions to produce the data are selected from (Hussain, et al., 2002)

1. $f(x, y) = x\sin(y) + y\sin(x)$, $\quad -2\pi \le x, y \le 2\pi \quad$ (5.1)

2. $f(x, y) = x\sin(x) + y\sin(y)$, $\quad -2\pi \le x, y \le 2\pi \quad$ (5.2)

3. $f(x, y) = 3x^2 ye^{(5x^2+y-7)} + y^2 \sin(3x + y)$, $\quad -1 \le x, y \le 1 \quad$ (5.3)

4. $f(x, y) = \dfrac{1}{1+5x^2+5y^2}$, $\quad -1 \le x, y \le 1 \quad$ (5.4)

The next section shows the original functions and different metamodel techniques to approximate the relationship between the input and output.

5.2.1 The Original Function 5.1

$$f(x, y) = x\sin(y) + y\sin(x) \qquad -2\pi \le x, y \le 2\pi$$

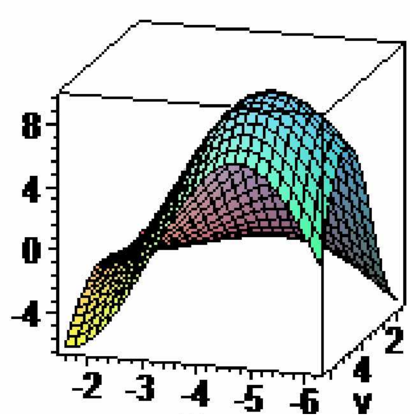


Figure 11. The original function 5.1

Input data points for the different metamodeling techniques are given in Table.10

Table 10: Actual Data Points for Bezier, RBF, and Polynomial regression

X	Y	F (X,Y)
-6.250	2.353	-4.354
-6.250	3.137	0.073
-6.250	3.920	4.518
-6.250	4.703	6.406
-6.250	5.487	4.650
`-6.25	6.270	0.290
-6.250	1.570	-6.198
-5.467	1.570	-4.323
-5.467	2.353	-2.162
-5.467	3.137	2.259
-5.467	3.920	6.695
-5.467	4.703	8.894
-5.467	5.487	7.907
-5.467	6.270	4.641
-4.683	1.570	-3.114
-4.683	2.353	-0.969
-4.683	3.137	3.112
-4.683	3.920	7.207
-4.683	4.703	9.384
-4.683	5.487	8.833
-4.683	6.270	6.329

X	Y	F(X,Y)
-3.900	1.570	-2.820
-3.900	2.353	-1.147
-3.900	3.137	2.138
-3.900	3.920	5.434
-3.900	4.703	7.135
-3.900	5.487	6.562
-3.900	6.270	4.364
-3.117	1.570	-3.156
-3.117	5.487	2.091
-3.117	6.270	-0.115
-3.117	2.353	-2.269
-3.117	3.137	-0.094
-3.117	3.920	2.091
-3.117	4.703	2.999
-2.333	2.353	-3.356
-2.333	3.137	-2.280
-2.333	3.920	-1.196
-2.333	4.703	-1.068
-2.333	5.487	-2.299
-2.333	6.270	-4.503

5.2.2 Bezier Fit for 5.1

We employed a triangular Bezier patch in MAPLE 8.0 to the bivariate output model 5.1. The result of bivariate output modeling using triangular Bezier patch is shown below.

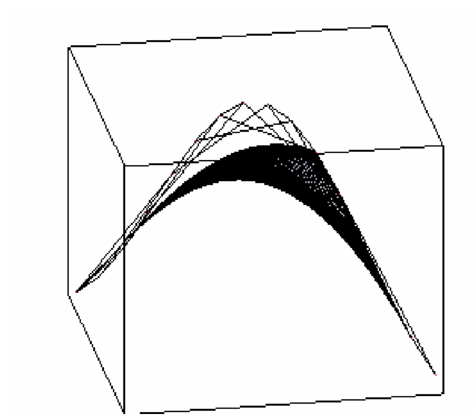


Figure 12. Bezier fit to the function 5.1

Figure 12 Shows the Bezier fit to the test points given in Table 10. Due to scaling constrained in Maple it looks like a 2-D diagram.

5.2.3 Polynomial Regression Fit for 5.1

We employed a degree three polynomial regression in MAPLE 8.0 to the bivariate output model 5.1. The result of bivariate output modeling using polynomial regressions is shown below.

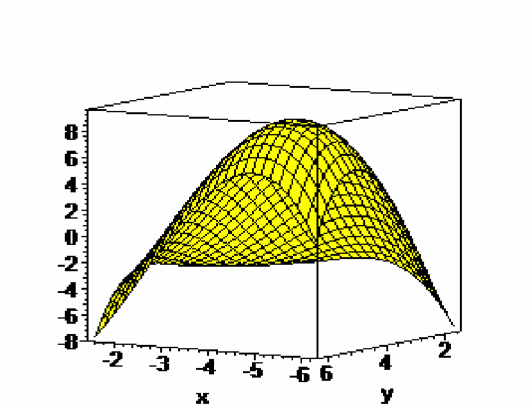


Figure 13: Polynomial regression fit to the function 5.1

Figure 13 Shows a polynomial regression fit to the test points given in Table 10.

5.2.4 Radial Basis Function Fit for 5.1

We employed a Multiquadratic and thin plate spline interpolation as a basis functions for RBF in MAPLE 8.0 to the bivariate output model 5.1. The result of bivariate output modeling using Multiquadratic and thin plate spline interpolations is shown below.

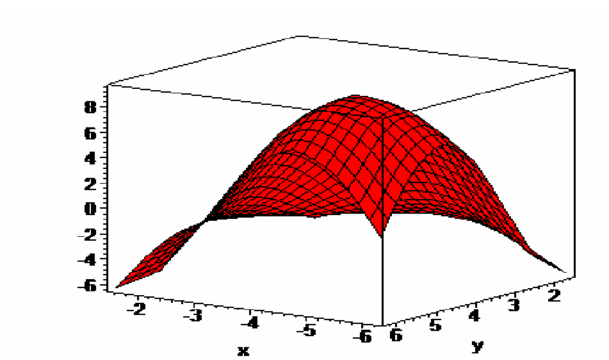


Figure 14: RBF using Multiquadratic fit to the function 5.1

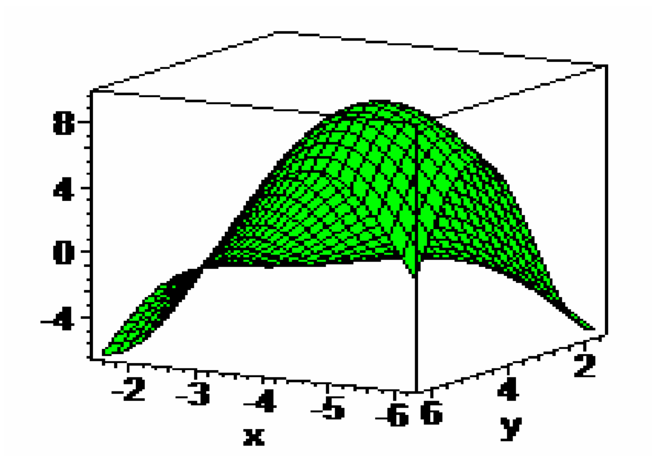


Figure15: RBF using thin plate spline fit to the function 5.1

Figure 14 and Figure 15 shows a RBF using Multiquadratic and thin plate spline Interpolation fit to the test points given in Table 10. Both MQ as well as TPS appear to be similar but the result of the interpolation shows that MQ performs better than TPS in all the four cases.

5.2.5 The Original Function 5.2

$$f(x, y) = x\sin(x) + y\sin(y)\text{, where } -2\pi \le x, y \le 2\pi$$

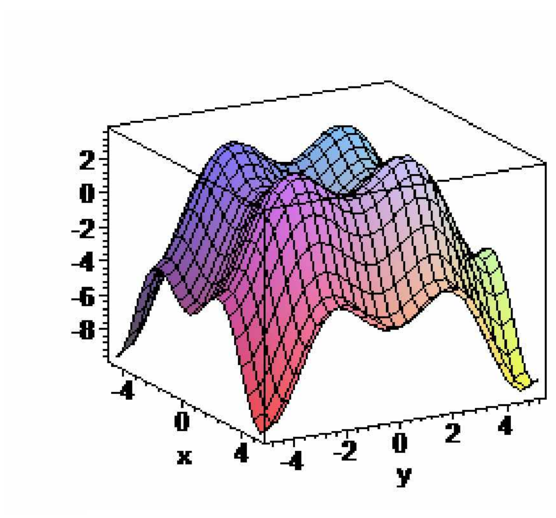


Figure 16: The original function 5.2

Input data points for the different metamodeling techniques are given in Table.11

Table11: Actual Data Points for Bezier, RBF, and Polynomial regression

X	Y	F (X,Y)
-4.913	-4.913	-9.629
-4.913	-3.163	-4.882
-4.913	-1.413	-3.419
-4.913	0.337	-4.703
-4.913	2.087	-2.999
-4.913	3.837	-7.273
-4.913	5.587	-8.397
-3.163	-4.913	-4.882
-3.163	-3.163	-0.135
-3.163	-1.413	1.328
-3.163	0.337	0.044
-3.163	2.087	1.747
-3.163	3.837	-2.526
-3.163	5.587	-3.651
-1.413	-4.913	-3.419
-1.413	-3.163	1.328
-1.413	-1.413	2.791
-1.413	0.337	1.507
-1.413	2.087	3.211
-1.413	3.837	-1.063
-1.413	5.587	-2.187

X	Y	F(X,Y)
0.337	-4.913	-4.703
0.337	-3.163	0.044
0.337	-1.413	1.507
0.337	0.337	0.223
0.337	2.087	1.926
0.337	3.837	-2.347
0.337	5.587	-3.471
2.087	-4.913	-2.999
2.087	-3.163	1.747
2.087	-1.413	3.211
2.087	0.337	1.926
2.087	2.087	3.630
2.087	3.837	-0.643
2.087	5.587	-1.768
3.837	-4.913	-7.273
3.837	-3.163	-2.526
3.837	-1.413	-1.063
3.837	0.337	-2.347
3.837	2.087	-0.643
3.837	3.837	-4.917
3.837	5.587	-6.041
5.587	-4.913	-8.397

5.2.6 Bezier Fit for 5.2

We employed a triangular Bezier patch in MAPLE 8.0 to the bivariate output model 5.2. The result of bivariate output modeling using triangular Bezier patch is shown below.

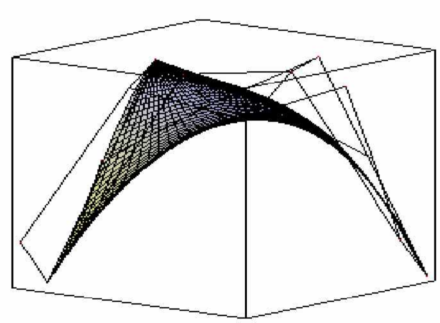


Figure 17: Bezier fit to the function 5.2

Figure 17 Shows the Bezier fit to the test points given in Table 11. Due to scaling constrained in Maple it looks like a 2-D diagram.

5.2.7 Polynomial Regression Fit for 5.2

We employed a degree three polynomial regression in MAPLE 8.0 to the bivariate output model 5.2. The result of bivariate output modeling using polynomial regressions is shown below.

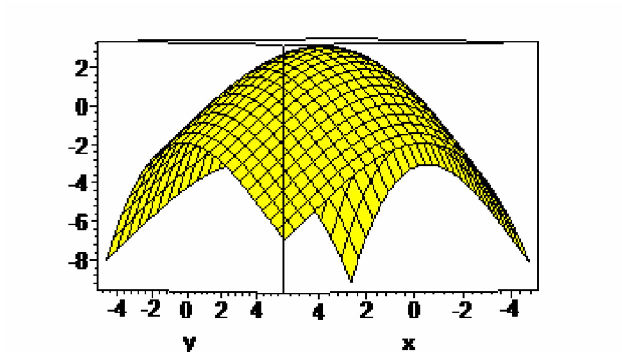


Figure18: Polynomial regression fit to the function 5.2

Figure 18. Shows a polynomial regression fit to the test points given in Table 11.

5.2.8 Radial Basis Function Fit for 5.2

We employed a Multiquadratic and thin plate spline interpolation as a basis functions for RBF in MAPLE 8.0 to the bivariate output model 5.2. The result of bivariate output modeling using Multiquadratic and thin plate spline interpolations is shown below.

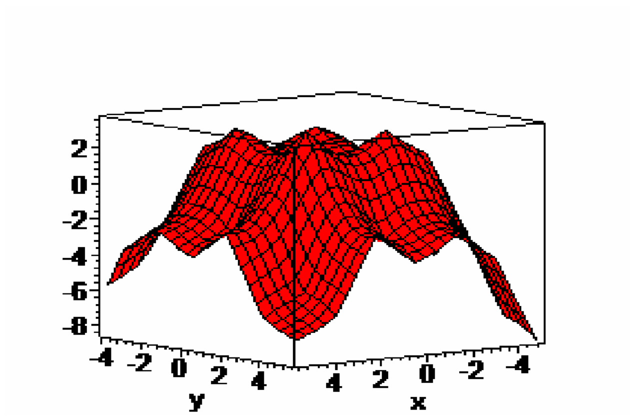


Figure 19: RBF using Multiquadratic fit to the function 5.2

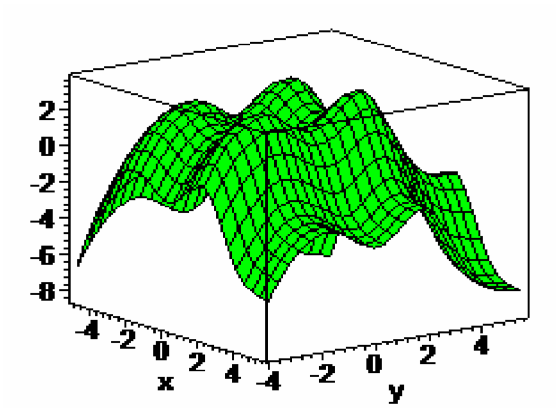


Figure20: RBF using thin plate spline fit to the function 5.1

Figure 19 and Figure 20 Shows a RBF using Multiquadratic and thin plate spline Interpolation fit to the test points given in Table 11.

5.2.9 The Original Function 5.3

$$f(x,y) = 3x^2 y e^{(5x^2+y-7)} + y^2 \sin(3x+y) \qquad -1 \le x, y \le 1$$

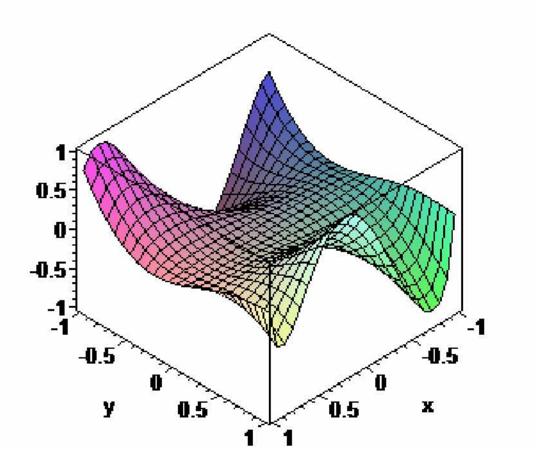


Figure 21: The original function 5.3

Input data points for the different metamodeling techniques are given in Table.12

Table 12: Actual Data Points for Bezier, RBF, and Polynomial regression

X	Y	F(x,y)
0.000	-0.667	0.310
-1.000	-1.000	0.091
-1.000	-0.667	0.122
-1.000	-0.333	0.153
-1.000	0.000	0.167
-1.000	0.333	0.153
-1.000	0.667	0.122
-1.000	1.000	0.091
-0.667	-1.000	0.122
-0.667	-0.667	0.184
-0.667	-0.333	0.265
-0.667	0.000	0.310
-0.667	0.333	0.265
-0.667	0.667	0.184
-0.667	1.000	0.122
-0.333	-1.000	0.153
-0.333	-0.667	0.265
-0.333	-0.333	0.474
-0.333	0.000	0.643
-0.333	0.333	0.474
-0.333	0.667	0.265

X	Y	F(X,Y)
-0.333	1.000	0.153
0.000	-1.000	0.167
0.000	-0.333	0.643
0.000	0.000	1.000
0.000	0.333	0.643
0.000	0.667	0.310
0.000	1.000	0.167
0.333	-1.000	0.153
0.333	-0.667	0.265
0.333	-0.333	0.474
0.333	0.000	0.643
0.333	0.333	0.474
0.333	0.667	0.265
0.333	1.000	0.153
0.667	-1.000	0.122
0.667	-0.667	0.184
0.667	-0.333	0.265
0.667	0.000	0.310
0.667	0.333	0.265
0.667	0.667	0.184
0.667	1.000	0.122
1.000	-1.000	0.091

5.2.10 Bezier Fit for 5.3

We employed a triangular Bezier patch in MAPLE 8.0 to the bivariate output model 5.3. The result of bivariate output modeling using triangular Bezier patch is shown below.

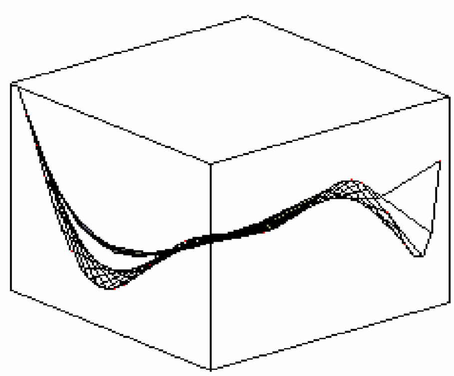


Figure 22: Bezier fit to the function 5.3

Figure 22 shows the Bezier fit to the test points given in Table 12. Due to scaling constrained in Maple it looks like a 2-D diagram.

5.2.11 Polynomial Regression Fit for 5.3

We employed a degree three polynomial regression in MAPLE 8.0 to the bivariate output model 5.3. The result of bivariate output modeling using polynomial regressions is shown below.

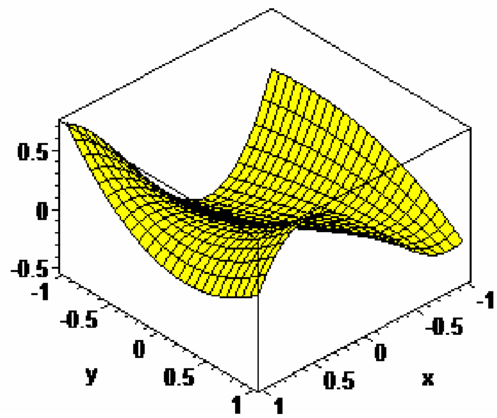


Figure 23: Polynomial regression fit to the function 5.3

Figure 23 shows a polynomial regression fit to the test points given in Table 12.

5.2.12 Radial Basis Function Fit for 5.2

We employed a Multiquadratic and thin plate spline interpolation as a basis functions for RBF in MAPLE 8.0 to the bivariate output model 5.3. The result of bivariate output modeling using Multiquadratic and thin plate spline interpolations is shown below.

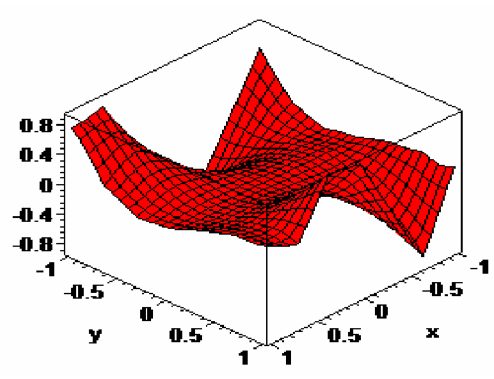


Figure24: RBF using Multiquadratic fit to the function

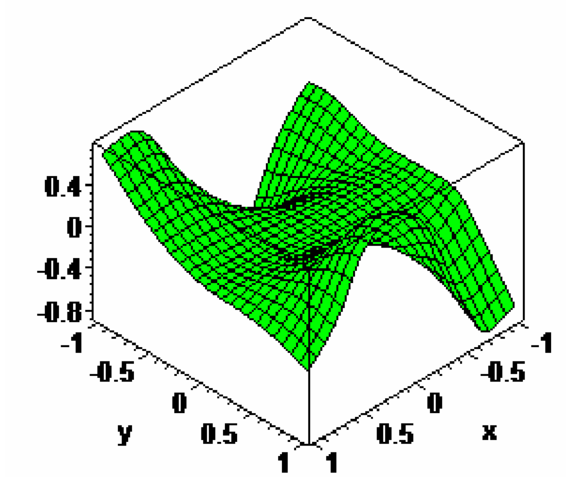


Figure 25: RBF using thin plate spline fit to the function 5.3

Figure 23 and Figure 24 Shows a RBF using Multiquadratic and thin plate spline Interpolation fit to the test points given in Table 12.

5.2.13 The Original Function 5.4

$$f(x,y) = \frac{1}{1+5x^2+5y^2} \qquad -1 \le x, y \le 1$$

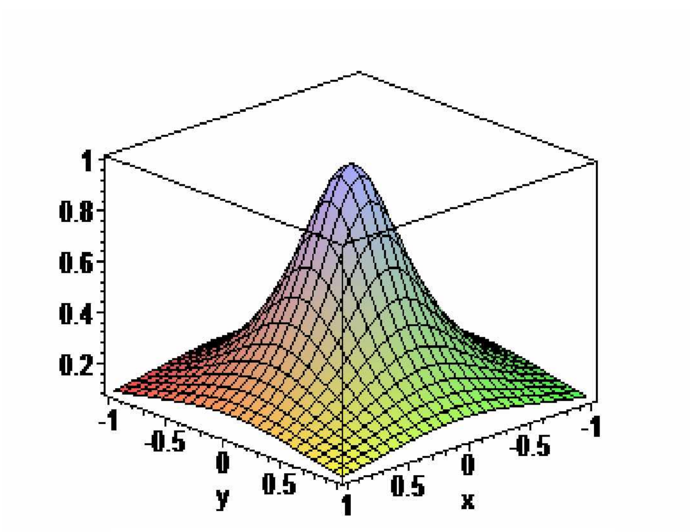


Figure 26: The original function 5.4

Input data points for the different metamodeling techniques are given in Table.13

Table 13: Actual Data Points for Bezier, RBF, and Polynomial regression

X	Y	F (X,Y)
0.333	0.333	0.108
-1.000	-1.000	0.607
-1.000	-0.667	0.084
-1.000	-0.333	-0.076
-1.000	0.000	0.000
-1.000	0.333	0.138
-1.000	0.667	0.206
-1.000	1.000	0.194
-0.667	-1.000	-0.145
-0.667	-0.667	-0.207
-0.667	-0.333	-0.083
-0.667	0.000	0.000
-0.667	0.333	-0.105
-0.667	0.667	-0.417
-0.667	1.000	-0.811
-0.333	-1.000	-0.909
-0.333	-0.667	-0.443
-0.333	-0.333	-0.108
-0.333	0.000	0.000
-0.333	0.333	-0.068
-0.333	0.667	-0.145

X	Y	F(X,Y)
-0.333	1.000	0.001
0.000	-1.000	-0.841
0.000	-0.667	-0.275
0.000	-0.333	-0.036
0.000	0.000	0.000
0.000	0.333	0.036
0.000	0.667	0.275
0.000	1.000	0.841
0.333	-1.000	0.000
0.333	-0.667	0.145
0.333	-0.333	0.069
0.333	0.000	0.000
0.333	0.667	0.443
0.333	1.000	0.911
0.667	-1.000	0.837
0.667	-0.667	0.428
0.667	-0.333	0.108
0.667	0.000	0.000
0.667	0.333	0.086
0.667	0.667	0.218
0.667	1.000	0.172
1.000	-1.000	0.760

5.2.14 Bezier Fit for 5.4

We employed a triangular Bezier patch in MAPLE 8.0 to the bivariate output model 5.4. The result of bivariate output modeling using triangular Bezier patch is shown below.

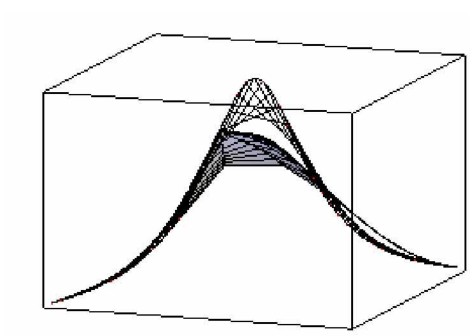


Figure 27: Bezier fit to the function 5.4

Figure 27 shows the Bezier fit to the test points given in Table 13. Due to scaling constrained in Maple it looks like a 2-D diagram.

5.2.15 Polynomial Regression Fit for 5.3

We employed a degree three polynomial regression in MAPLE 8.0 to the bivariate output model 5.4. The result of bivariate output modeling using polynomial regressions is shown below

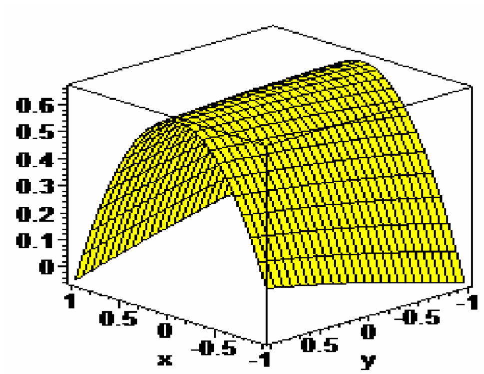


Figure 28: Polynomial regression fit to the function 5.4

Figure 28 shows a polynomial regression fit to the test points given in Table 13.

5.2.16 Radial Basis Function Fit 5.4

We employed a Multiquadratic and thin plate spline interpolation as a basis functions for RBF in MAPLE 8.0 to the bivariate output model 5.4. The result of bivariate output modeling using Multiquadratic and thin plate spline interpolations is shown below.

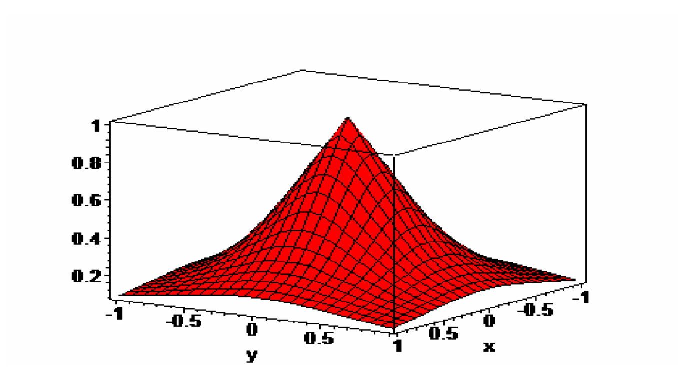


Figure29: RBF using Multiquadratic fit to the function 5.4

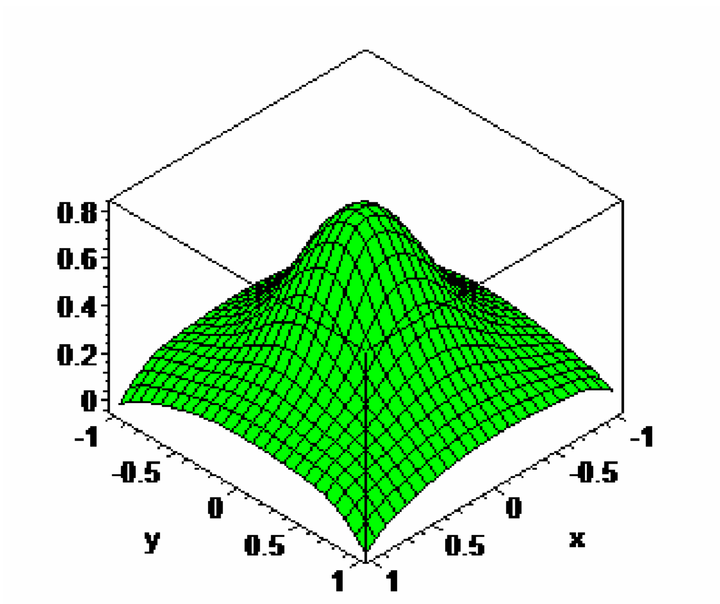


Figure 30: RBF using thin plate spline fit to the function 5.3

Figure 29 and Figure 30 shows a RBF using Multiquadratic and thin plate spline Interpolations fit to the test points given in Table 13.

5.3 Validation of Bivariate Models

The outcomes of the simple Bezier triangular patch is validated via comparison with other metamodeling techniques like radial basis function (RBF) and polynomial regression (PR). To access the performance of results, three different metrics are used: R square, relative average absolute error, and relative maximum absolute error. The robustness of result is also measured by means of comparison of sum of squared error of results and polynomial regression and radial basis functions. The next section shows the outcomes and result of comparison between different metamodeling techniques for the random points. The outcomes from the bivariate model (5.1, 5.2, 5.3, and 5.4) using triangular Bezier patch, radial basis function, and polynomial regression is shown in Tables 14, 15, 16,1and 17, and Tables 14.a, 15.a, 16.a and 17.a compare the performance matrix of each metamodeling technique.

Table 14: Outcome of different metamodeling techniques for 5.1

x	y	$f(x,y)$	$f_B(x,y)$	$f_{R1}(x,y)$	$f_{R2}(x,y)$	$f_{PR}(x,y)$
-6.052	1.768	-5.530	-5.245	-5.352	-5.207	-5.609
-5.916	1.904	-4.907	-4.539	-4.709	-4.581	-4.324
-5.714	2.106	-3.782	-3.487	-3.631	-3.585	-2.579
-5.548	2.272	-2.715	-2.647	-2.655	-2.658	-1.280
-5.369	2.451	-1.477	-1.791	-1.460	-1.507	-0.023
-5.041	2.779	0.841	-0.406	0.841	0.794	1.892
-4.753	3.067	2.710	0.583	2.701	2.700	3.170
-4.361	3.459	4.610	1.544	4.518	4.608	4.303
-4.108	3.712	5.273	1.912	5.188	5.270	4.664
-3.842	3.978	5.416	2.077	5.385	5.415	4.734
-3.468	4.352	4.640	1.913	4.548	4.638	4.294
-3.250	4.570	3.712	3.712	3.661	3.713	3.750
-3.154	4.666	3.208	3.147	3.193	3.210	3.444
-3.010	4.810	2.731	2.262	2.726	2.725	2.907
-3.070	4.750	2.365	1.223	2.354	2.349	3.144
-2.968	4.852	2.098	0.956	2.087	2.076	2.731
-2.914	4.906	1.753	1.650	1.742	1.722	2.498
-2.818	5.002	1.110	1.022	1.106	1.068	2.049
-2.757	5.063	0.692	0.297	0.695	0.645	1.744
-2.674	5.146	0.107	0.063	0.120	0.057	1.298
-2.650	5.170	-0.063	-0.098	-0.047	-0.113	1.164
-2.530	5.290	-0.918	-0.910	-0.894	-0.962	0.457
-2.266	5.554	-2.755	-2.693	-2.703	-2.709	-1.323
-2.170	5.650	-3.382	-3.332	-3.266	-3.255	-2.045
-2.074	5.746	-3.972	-3.959	-3.811	-3.764	-2.808
-1.978	5.842	-4.520	-4.572	-4.329	-4.240	-3.611
-1.882	5.938	-5.016	-5.168	-4.811	-4.686	-4.454
-1.810	6.010	-5.351	-5.602	-5.150	-5.009	-5.113
-1.762	6.058	-5.554	-5.884	-5.366	-5.222	-5.565

Table 14a: Performance of different metamodeling techniques for 5.1

	Bezier	MQ	TPS	PR
R-square	0.878927	0.999202	0.997994	0.937634
RAAE	0.192736	0.019522	0.028157	0.211366
RMAE	0.899889	0.054797	0.093959	0.389106

R square $_{\mathbf{(Bezier)}}$ < R square $_{\mathbf{(RBF\_MQ)}}$ < R square $_{\mathbf{(RBF\_TPS)}}$ > R square $_{(PR\_3)}$,

RAAE $_{\mathbf{(Bezier)}}$ > RAAE $_{\mathbf{(RBF\_MQ)}}$ > RAAE $_{\mathbf{(RBF\_TPS)}}$ < RAAE $_{\mathbf{(PR\_3)}}$, and

RMAE $_{\mathbf{(Bezier)}}$ > RMAE $_{\mathbf{(RBF\_MQ)}}$ > RMAE $_{\mathbf{(RBF\_TPS)}}$ < RMAE $_{\mathbf{(PR\_3)}}$

Table 15: Outcome of different metamodeling techniques for 5.2

x	y	$f(x,y)$	$f_B(x,y)$	$f_{R1}(x,y)$	$f_{R2}(x,y)$	$f_{PR}(x,y)$
-4.459	-4.459	-8.633	-7.607	-7.044	-6.697	-6.642
-4.119	-4.119	-6.827	-6.252	-5.141	-4.678	-4.893
-3.748	-3.748	-4.271	-4.946	-3.139	-2.707	-3.212
-3.135	-3.135	0.041	-3.169	-0.053	-0.029	-0.917
-2.916	-2.916	1.307	-2.643	0.523	0.751	-0.235
-2.212	-2.212	3.545	-1.320	1.995	2.458	1.490
-1.441	-1.441	2.857	-0.445	2.778	2.815	2.635
-0.888	-0.888	1.379	-0.145	2.069	1.858	3.023
-0.601	-0.601	0.680	-0.084	1.676	1.275	3.096
-0.003	-0.003	0.000	-0.150	0.832	0.339	2.990
1.282	1.282	2.459	-1.043	2.344	2.594	1.755
2.423	2.423	3.190	3.630	2.085	2.572	0.432
2.591	2.591	2.711	2.606	1.321	1.820	-0.215
2.759	2.759	2.060	1.952	0.558	0.987	-0.906
2.983	2.983	0.942	1.219	-0.479	-0.214	-1.387
3.095	3.095	0.288	0.138	-1.012	-0.842	-1.634
3.207	3.207	-0.419	-0.439	-1.559	-1.483	-1.883
3.431	3.431	-1.958	-1.036	-2.699	-2.778	-2.391
3.599	3.599	-3.179	-2.272	-3.597	-3.727	-2.778
3.823	3.823	-4.816	-3.222	-4.848	-4.857	-3.302
3.991	3.991	-5.994	-4.494	-5.131	-5.449	-3.698
4.103	4.103	-6.729	-5.434	-5.289	-5.757	-3.964
4.327	4.327	-8.019	-7.217	-5.628	-6.247	-4.497
4.719	4.719	-9.438	-9.003	-6.214	-6.788	-5.430
5.111	5.111	-9.421	-10.285	-6.695	-7.007	-6.354
5.279	5.279	-8.908	-10.630	-6.870	-7.048	-6.744
5.503	5.503	-7.742	-10.860	-7.085	-7.117	-7.256
5.671	5.671	-6.518	-10.837	-7.297	-7.242	-7.633
5.839	5.839	-5.018	-10.627	-7.564	-7.448	-8.003

Table 15: Performance of different metamodeling techniques for 5.2

	Bezier	**MQ**	**TPS**	**PR**
R-square	0.774361	0.905725	0.914903	0.759711
RAAE	0.32408	0.244848	0.233697	0.424702
RMAE	1.207577	0.694017	0.592954	0.862788

R square $_{\textbf{(Bezier)}}$ < R square $_{\textbf{(RBF\_MQ)}}$ < R square $_{\textbf{(RBF\_TPS)}}$ >R square $_{\text{(PR\_3)}}$,

RAAE $_{\textbf{(Bezier)}}$ > RAAE $_{\textbf{(RBF\_MQ)}}$ > RAAE $_{\textbf{(RBF\_TPS)}}$ < RAAE $_{\textbf{(PR\_3)}}$, and

RMAE $_{\textbf{(Bezier)}}$ > RMAE $_{\textbf{(RBF\_MQ)}}$ > RMAE $_{\textbf{(RBF\_TPS)}}$ > RMAE $_{\textbf{(PR\_3)}}$

Table 16: Outcome of different metamodeling techniques for 5.3

x	y	$f(x,y)$	$f_B(x,y)$	$f_{R1}(x,y)$	$f_{R2}(x,y)$	$f_{PR}(x,y)$
-0.922	-0.922	0.381	0.393	0.351	0.325	0.289
-0.878	-0.878	0.241	0.289	0.215	0.182	0.198
-0.827	-0.827	0.093	0.190	0.077	0.041	0.108
-0.809	-0.809	0.044	0.159	0.032	-0.003	0.079
-0.707	-0.707	-0.159	0.025	-0.159	-0.173	-0.050
-0.662	-0.662	-0.211	-0.017	-0.211	-0.210	-0.091
-0.452	-0.452	-0.199	-0.096	-0.198	-0.182	-0.167
-0.392	-0.392	-0.154	-0.096	-0.156	-0.145	-0.160
-0.330	-0.330	-0.106	-0.090	-0.106	-0.106	-0.140
-0.265	-0.265	-0.061	-0.077	-0.073	-0.067	-0.109
-0.197	-0.197	-0.028	-0.060	-0.042	-0.035	-0.066
-0.090	-0.090	-0.003	-0.029	-0.012	-0.008	0.018
-0.015	-0.015	0.000	-0.005	-0.002	-0.001	0.086
0.167	0.167	0.017	0.017	0.029	0.023	0.265
0.190	0.190	0.025	0.026	0.037	0.031	0.288
0.208	0.208	0.032	0.034	0.044	0.038	0.305
0.295	0.295	0.081	0.084	0.087	0.084	0.388
0.301	0.301	0.085	0.088	0.090	0.087	0.393
0.377	0.377	0.142	0.142	0.138	0.137	0.459
0.400	0.400	0.161	0.160	0.154	0.152	0.477
0.429	0.429	0.183	0.182	0.172	0.171	0.499
0.511	0.511	0.235	0.240	0.212	0.212	0.552
0.575	0.575	0.251	0.275	0.226	0.228	0.583
0.604	0.604	0.249	0.286	0.227	0.229	0.594
0.668	0.668	0.216	0.294	0.218	0.217	0.609
0.709	0.709	0.175	0.285	0.222	0.201	0.611
0.779	0.779	0.074	0.236	0.227	0.177	0.603
0.814	0.814	0.016	0.193	0.233	0.175	0.591
0.884	0.884	-0.073	0.063	0.261	0.203	0.554
0.919	0.919	-0.067	-0.026	0.283	0.235	0.527
0.966	0.966	0.071	-0.173	0.319	0.294	0.482

Table 16a: Performance of different metamodeling techniques for 5.3

	Bezier	MQ	TPS	PR
R-square	0.711694	0.602633	0.582504	0.2437
RAAE	0.347245	0.2897	0.325984	0.491955
RMAE	1.40276	2.011481	2.064926	2.8347

R square $_{\textbf{(Bezier)}}$ > R square $_{\textbf{(RBF\_MQ)}}$ > R square $_{\textbf{(RBF\_TPS)}}$ > R square $_{\text{(PR\_3)}}$,

RAAE $_{\textbf{(Bezier)}}$ > RAAE $_{\textbf{(RBF\_MQ)}}$ > RAAE $_{\textbf{(RBF\_TPS)}}$ < RAAE $_{\textbf{(PR\_3)}}$, and

RMAE $_{\textbf{(Bezier)}}$ > RMAE $_{\textbf{(RBF\_MQ)}}$ > RMAE $_{\textbf{(RBF\_TPS)}}$ < RMAE $_{\textbf{(PR\_3)}}$

Table 17:Outcomes of different metamodeling techniques for 5.4

x	y	$f(x,y)$	$f_B(x,y)$	$f_{R1}(x,y)$	$f_{R2}(x,y)$	$f_{PR}(x,y)$
-0.971	-0.971	0.096	0.097	0.096	0.102	-0.009
-0.922	-0.922	0.105	0.109	0.106	0.116	0.057
-0.845	-0.845	0.123	0.135	0.124	0.133	0.153
-0.827	-0.827	0.128	0.142	0.129	0.137	0.174
-0.770	-0.770	0.144	0.169	0.145	0.150	0.238
-0.707	-0.707	0.167	0.204	0.167	0.169	0.304
-0.662	-0.662	0.186	0.234	0.186	0.186	0.348
-0.614	-0.614	0.210	0.268	0.213	0.209	0.392
-0.536	-0.536	0.258	0.328	0.266	0.257	0.455
-0.481	-0.481	0.302	0.374	0.312	0.302	0.495
-0.362	-0.362	0.433	0.475	0.438	0.434	0.566
-0.231	-0.231	0.651	0.579	0.631	0.650	0.620
-0.090	-0.090	0.925	0.664	0.861	0.908	0.653
-0.015	-0.015	0.998	0.689	0.984	0.995	0.658
0.167	0.167	0.783	0.783	0.736	0.773	0.639
0.184	0.184	0.747	0.747	0.707	0.740	0.635
0.202	0.202	0.711	0.712	0.679	0.706	0.630
0.219	0.219	0.676	0.678	0.651	0.673	0.625
0.295	0.295	0.535	0.542	0.531	0.536	0.597
0.313	0.313	0.506	0.513	0.504	0.507	0.589
0.383	0.383	0.406	0.411	0.414	0.406	0.555
0.429	0.429	0.352	0.354	0.362	0.352	0.528
0.493	0.493	0.291	0.289	0.301	0.291	0.486
0.517	0.517	0.273	0.268	0.281	0.272	0.470
0.552	0.552	0.247	0.240	0.254	0.246	0.443
0.604	0.604	0.215	0.205	0.219	0.214	0.400
0.645	0.645	0.194	0.182	0.195	0.193	0.364
0.703	0.703	0.168	0.155	0.169	0.170	0.309
0.762	0.762	0.147	0.133	0.148	0.153	0.248
0.803	0.803	0.134	0.120	0.136	0.143	0.203
0.861	0.861	0.119	0.105	0.120	0.130	0.134

Table 17a: Performance of different metamodeling techniques for 5.4

	Bezier	**MQ**	**TPS**	**PR**
R-square	0.918611	0.995393	0.999453	0.862394
RAAE	0.128687	0.036177	0.01613	0.491955
RMAE	1.178164	0.24593	0.063962	1.296308

R square $_{\mathbf{(Bezier)}}$ < R square $_{\mathbf{(RBF\_MQ)}}$ < R square $_{\mathbf{(RBF\_TPS)}}$ >R square $_{(PR\_3)}$,

RAAE $_{\mathbf{(Bezier)}}$ > RAAE $_{\mathbf{(RBF\_MQ)}}$ > RAAE $_{\mathbf{(RBF\_TPS)}}$ < RAAE $_{\mathbf{(PR\_3)}}$, and

RMAE $_{\mathbf{(Bezier)}}$ > RMAE $_{\mathbf{(RBF\_MQ)}}$ > RMAE $_{\mathbf{(RBF\_TPS)}}$ > RMAE $_{\mathbf{(PR\_3)}}$

As shown in above Tables every model is tested quantitatively; where x and y are independent variable, and $f(x,y)$, $f_B(x,y)$, $f_{R1}(x,y)$, $f_{R2}(x,y)$,and $f_{PR}(x,y)$ represents the original function ,Bezier Fit , Radial basis function with multiquardic (MQ) interpolation ,Radial basis function with thin plate spline (TPS) interpolation , and polynomial regression (PR) respectively. The curve is fitted to the test point by varying the value of s , t and, u in equation

$$P(s,t,u) = \sum_{i+j+,k=n} T_{i,j,k} B_{i,j,k}(s,t,u)$$

in such a way that $r+s+t=1$. Similarily, the RBF which performed better than all other method fitted using MQ and TPS functions

$$MQ(x,y) = \sum_{j=1}^{K} \beta_j \left\| x - x_{ej} \right\| \left\| y -_{ej} \right\|,$$

$$TPS(x,y) = \sum \beta_j \left[\left\| x - x_{ej} \right\| \left\| y - y_{ej} \right\| \right]^2 \ln \left[\left\| x - x_{ej} \right\| \left\| y - y_{ej} \right\| \right]$$

by calculating the coefficient β_j for each pair of input.

The SAS software package is used to approximate all the functions for polynomial regression using PROC REG:

For function 5.1

$f_{PR}(x,y)$ = 27.76692 + 17.33275 x -17.08885 y + 3.82572 x^2 + 3.75247 y^2 + 0.32698 x^3 -0.31909 y^3 -3.85655 xy -0.20387 $x^2 y$ + 0.20813 $y^2 x$

For function 5.2

$f_{PR}(x, y)$ = 2.988 -0.2274 x -0.22 y -0.22 x^2 + 0.0166 y^2 +0.0166 x^3 + -0.31909 y^3 -2.65 E-17 xy -2.594E-18 $x^2 y$ + 1.562E-17 $y^2 x$

For function 5.3

$f_{PR}(x, y)$ = 0.1109 + 0.5712 x + 0.3826 y + 0.21938 x^2 +0.1416 y^2 -0.6861 x^3 -0.01678 y^3 +3.8E-11 xy -0.607 $x^2 y$ + 0.338 $y^2 x$

For function 5.4

$f_{PR}(x, y)$ =0.65845+0.000000323 x +0.000502 y -0.604383 x^2 -0.093032 xy + -0.010383 y^2

In function 5.4, the higher degree polynomial decreased the value of R-square therefore it is limited to only 2 degree polynomial.

Every metamodel generated in this study is tested qualitatively as well as quantitatively. The quantitative test is conducted with the help of comparing R-square, Relative average absolute error (RAAE), and Relative maximum absolute error (RMAE). The qualitative test is conducted by doing a visual comparison of metamodels with the true function. The performance matrix is computed with the 30 random points generated in the region.

6 Conclusions

In this thesis two different models; univariate model and bivariate model is generated with different metamodels like radial basis function and polynomial regression.

In the preliminary work, univariate model with Bezier fit was compared with polynomial metamodel and found to be best suited method for simulation metamodels. The Bezier fit metamodel is implemented in MAPLE 8.0. Successful result of univariate metamodel motivated to develop a Bezier fit bivariate metamodel.

One of the advantages of metamodels is that it yield close relationship between performance measure (output) and design parameters (input). Metamodeling also provides a fast, reasonably priced, and computationally efficient analysis tool for optimization and design space investigation to identify the key variables. The most significant weakness of metamodel is the lack of availability of software for metamodeling techniques like radial basis function, Multivariate adaptive regression spline metamodel ,and artificial neural network metamodel, moreover the application of metamodel based on the principal of interpolation are also restricted.

The second part of the work discussed bivariate model. The Bezier fit, radial basis function, and polynomial metamodels are generated. The Bezier fit and radial basis function metamodel are implemented in MAPLE and Polynomial metamodel is implemented is SAS. The model generated with Bezier fit is verified and tested quantitatively and qualitatively.

1. Based on the test results in Section 5. The following conclusions can be made:

- Bezier fit metamodels provided a better fit in function 5.3, in all the other cases radial basis function model provided a better fit than polynomial metamodels and Bezier fit metamodels.

- A Bezier fit metamodels provided a better fit than the polynomial metamodels in all the cases.
- Software for automated computation assessment for polynomial models are in abundances, whereas similar software does not exist for radial basis function and Bezier metamodels. With the help of MAPLE and MATLAB the computation of metamodels using radial basis function and Bezier fit can be achieve without great difficulty.

2. Based on the result obtained from the RAAE the following could be inferred :

a. A Bezier metamodels provided a better fit than the radial basis function and polynomial metamodels using small number of design points.

b. Even R-square of polynomial metamodels are greater than the Bezier fit metamodels but the deviation from the original function is greater for polynomial metamodels as compare to the Bezier fit metamodels.

3. Based on the computations the following could be inferred:

a. A Bezier metamodel are easier to implement as compare to the radial basis function.

b. The input points required to generate the Bezier metamodel output are less as compare to radial basis function and polynomial regression.

- **Future Work**

This study was limited to the response function of two variables only, and did not include actual simulation process. Future studies should examine the ability of Bezier fit on broad class of simulation. In Section 5 Bezier fit has performed better than the radial basis function in 5.3, further investigation must be done in this area to find the compatibility of the Bezier fit.

References

Barton, R. R., 1998, "Simulation metamodels," Proceedings of the 1998 Winter Simulation Conference, pp.167-174

Barton, R. R., 1992, "Metamodels for simulation input-output Relations," proceedings of the 1992 Winter Simulation Conference, pp. 289-299

Cressie, N. A. C., 1993, *Statistics for spatial data,* New York: John Wiley & Sons

Franke, Richard, 1982, "Scattered data interpolation: tests of some methods," *Mathematics of Computation 38,* pp.181-200

Gerald, F., 2000, *Curves and surfaces for CAGD: A Practical Guide*, Morgan Kaufmann Publishers, pp. 81-93

Gupta, P. K., Hira, D. S., 1999,*Operation research*, New Delhi: S .Chand & Company

Hussain, M.F, Joshi, S., Barton, R. R,2002, "Metamodeling: radial basis functions versus polynomials," *European Journal of Operational Research 138*, 142-154

Jin, R., Chen, W., Simpson, T. W., 2000, "Comparative studies of metamodeling techniques under multiple modeling criteria," Proceedings of the 8th AIAA/USAF/NASA/ISSMO Multidisciplinary Analysis & Optimization Symposium, AIAA 2000-4801, Long Beach, CA

Koehler, J. R. and Owen, A. B., 1999, *Computer experiments*, Handbook of Statistics (Ghosh, and Rao, C. R., eds.), Elsevier Science, New York, pp. 261-308

Law, A. and Kelton, W., 1991, *Simulation modeling & analysis*, McGraw-Hill, New York, 2nd edition

Lionel, C., Bernd, F., Ferdinand, V., 2000, "Using multiple adaptive regression splines to understand trends in inspection data and identify optimal inspection rates," International Software Engineering Research Network (ISERN).

Mahdi, N., Mansooreh, M., 1999, **"**The development of a methodology for the use of neural networks and simulation modeling in system design," Proceedings of the 1999 Winter Simulation Conference, pp.537-542.

McAllister, C., Altuntas, B., Frank, M., and Potoradi, J., 2001, "Implementation of response surface methodology using variance reduction techniques in semiconductor manufacturing," proceedings of the Winter Simulation Conference 2001, pp. 1225-1230.

Myers, R. H., Montgomery, D.C., 1995, *Response surface methodology-process and product optimization using designed experiment*, New York: John Wiley & Sons.

Myers, R. H., Montgomery, D.C., 2002, *Response surface methodology-process and product optimization using designed experiment*, New York: John Wiley & Sons.

Rovenski, V., 2000, *Geometry of curves and surfaces with MAPLE*, Birkhauser Boston.

Sacks, J., Welch, W. J., Mitchell, T. J. and Wynn, H. P., 1989, "Design and analysis of computer experiments," *Statistical Science, Vol. 4,* No., pp. 409-435.

Simpson, T., Mauery, T., Korte, J., and Mistree, F., 1998, "Comparison of response surface and Kriging models For multidisciplinary design optimization," Technical Report AIAA 1998-4755.

Wagner, M., and Wilson, J., 1994, "Using bivariate Bezier distributions to model simulation input processes," Proceedings of the 1994 Winter Simulation Conference, pp.324 -331.

Wagner, M., and Wilson, J., 1996, "Recent developments in input modeling with Bezier distributions," Proceedings of the 1996 Winter Simulation Conference, pp.1448-1456.

Wagner, M., and Wilson, J., 1996. "Univariate Bezier distributions to model simulation input processes," *IIE Transactions 28 (9)*: pp. 699–712.

VITA

Harish Kingre was born in Nagpur, an old business and historical town in the state of Maharashtra, India. He is the son of Mrs. Darshna Kingre and Mr. Jivanlal Kingre. He got his primary and secondary education in Nagpur. He received the Bachelor of Science in mechanical engineering from Kavikulguru institute of technology and science, Nagpur in 2001. He worked as an intern in Grace Automotive private ltd. and Online Computer Institute. Later, he joined the graduate program at Louisiana State University, Baton Rouge (LSU) in fall, 2002. He worked in simulation, modeling, operation research, and experimental statistics area, during his graduate program at LSU. He is a candidate for the degree of Master of Science in Industrial Engineering to be awarded at the commencement of December 2004.