AN EXPLORATION AND EXPLOITATION PARETO APPROACH TO SURROGATE OPTIMIZATION

by

JOHN F DICKSON

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2014

I dedicate this dissertation to my parents for their unconditional love

and support

Acknowledgements

I want to thank my committee chair Dr. Victoria Chen for accepting me as her student and guiding me throughout my studies. Her knowledge on statistics inspired me to achieve and learn more. She helped me identify my weaknesses and provided feedback to help me improve. It's been a great honor for me to be a doctoral student under her. I also thank Dr. Jay Rosenberger for being my co-chair. His expertise in optimization helped develop my research methods to new heights. I enjoyed his mentorship thoroughly, and I am grateful to be his student. With both of my advisors' guidance I have made a transformation from a mechanical engineer to an analytics and operations research expert. Many thanks to Dr. John Priest for his valuable inputs. I also thank him for partially funding my Master's studies.

I have made some really good friends in the Center on Stochastic, Modeling, Optimization, & Statistics (COSMOS) whom have provided me with great support throughout my PhD studies. Together we were able to make COSMOS a fun place to study. Thanks to my department for continually funding my PhD studies. I also thank Dr. Susan Ferreira for supporting my first year in the PhD program.

I would like to thank my cousin Dr. Priscilla Roche and uncle Dr. Francis Roche for taking time to proof read my dissertation. I also thank my uncle and aunt, Dr. and Mrs. Roche for being my moral support throughout my days at UTA.

Finally, I extend my heartfelt thanks to my sister, my dad and my mom for always giving me their support. I could never imagine myself succeeding here without their help and guidance.

November 19, 2014

Abstract

AN EXPLORATION AND EXPLOITATION PARETO APPROACH TO SURROGATE OPTIMIZATION

John F Dickson, PhD

The University of Texas at Arlington, 2014

Supervising Professors: Victoria C. P. Chen and Jay M. Rosenberger

The experiments or simulations conducted by computers can be a tedious task, requiring substantial computational time. To find a global solution using a computer experiments process, we usually need to perform many function evaluations of the computer model. This research focuses on developing an optimization method to find a globally optimal solution efficiently using surrogates. The surrogates represent the computer model or the system that reads the inputs and generates the output responses of interest, so that these surrogate models can be used in place of time-consuming simulations runs. In surrogate based optimization, we iteratively build a surrogate model (a.k.a, approximate model or a metamodel) and conduct an optimization step, adding points in each iteration only as needed.

The proposed surrogate optimization method, Exploration and Exploitation Pareto Approach (EEPA), combines the notions of exploration and exploitation to seek the best solution with fewer function evaluations. Exploration is used to explore the points in an unexplored region. Exploration does not use the surrogate to look for new points. Four different exploration methods were used in this research, specifically maximin distance (Johnson, et. al., [43]), cosine (Corley et.al. [20]), Sobol′ (Sobol', [89])

sequence and Monte Carlo method (Niederreiter [56]). Maximin looks for points that are at maximin distance from the existing points, cosine looks for the maximum angular distance between points, Sobol' looks for points that are evenly spaced and Monte Carlo looks for points randomly in the input space.

Exploitation is used to explore promising areas in the input space. Exploitation requires a surrogate or metamodel to be built in order to look for new points. Different surrogate or metamodel models, including multivariate adaptive regression splines, radial basis functions, and treed regression, are used in this study. The minimum response method (Regis and Shoemaker [61]; Regis and Shoemaker [62] is used as the exploitation method in this study. Response metric helps to look for points that possibly give better solution.

Exploration and exploitation are combined in EEPA by obtaining a Pareto frontier. A Pareto frontier represents the non-dominating solutions given two or more solutions. The Pareto frontier is obtained by balancing the tradeoff between maximizing the exploration metric and minimizing the predicted response. . Points are then chosen from this Pareto frontier at which additional function evaluations using the computer model are executed.

Various test functions were used to compare EEPA to pure exploitation and pure exploration methods. In addition, a green building test function is also considered. These test functions are of different dimensions and structure. The green building data is simulated from a computer model called eQUEST. The results showed that EEPA reached the best solution faster than pure exploration or exploitation methods. Also, among the exploration methods, the cosine method performed well.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

1.1 Surrogate Optimization

Surrogate models, or metamodels, are approximate models for black-box functions, which are built using sampled data. *Surrogate optimization* is an optimization technique that makes use of surrogate models to find an optimal solution. In surrogate optimization, once a surrogate model is built, an optimal solution is then found using optimization algorithms such as genetic algorithm (Davis, [19]). Surrogate optimization evolved as a successor of Design of Experiments (DOE), Response Surface Methodology (RSM), and Design and Analysis of Computer Experiments (DACE). DOE is the classical method to gather inputs for experiments, at which will the outputs (responses) are observed (Jeffers [39])). RSM is an optimization technique for physical experiments with an assumption that, though the response surface is not perfect quadratic polynomial in given dimensions, it provides good approximation (Box and Wilson [9]). When computer models are used for experiments, DACE is used. It involves organizing input points using DOE, obtaining the output from a computer model, and then finally analyzing the data by building a metamodel (Sacks et.al [67]). No optimization is involved in DACE. DACE is sometimes performed sequentially when computer experiments are time consuming. Surrogate optimization performs the optimization over the metamodel created by DACE. This research employs a unique approach for surrogate optimization.

1.2 Research Motivation and Goal

The applications in which computer experiments or simulations are used as an engineering decision making tool has grown. Computer experiments can be time consuming, which might delay a decision-making process because all possible scenarios

have to be simulated and studied. Statistical and optimization techniques can help by optimally selecting the simulations to be performed knowing the importance of the engineering application.

The goal of this research is to develop a method by combining DACE and optimization to minimize the computer experiment runs to obtain the best solution.

1.3 Framework

Chapter 2 provides a literature review of surrogate optimization. Chapter 3 introduces EEPA and provides a detailed discussion of exploitation and exploration approaches to surrogate optimization. The computational study performed to demonstrate the proposed methodology is discussed in chapter 4. The results of the computational study are discussed in chapter 5. A tree based optimization model is discussed in detail in chapter 6, and finally conclusions and future research are discussed in chapter 7.

Chapter 2

Literature Review

2.1 Surrogate Optimization

Surrogate models used in different optimization frameworks in the past include response surface models, kriging models, radial basis functions, neural networks, and multivariate adaptive regression splines. This section previews some of the optimization frameworks developed on the surrogate models.

A recent study by Müller and Shoemaker in 2014 [53] proposed ensemble metamodels in the surrogate optimization process. This study involves two process, selection of metamodel combination and selection of candidate points. The Dempster–Shafer Theory [64] approach is used to combine metamodels and also provides weights for the metamodel used in the combination. The candidate points are selected using two criteria, distance and response surface with the help of a weighted score. A weight factor is assigned to these criteria for each of the candidate points. These weight factors are predefined and cycle through iteration until the best score is achieved. The candidate point is the one with the best weighted score.

Regis and Shoemaker [62] proposed a metric stochastic response surface (MSRS) method of global optimization to optimize a black box function that used the same method to select candidate points as discussed in Muller and Shoemaker [53] but with one metamodel, radial basis function. Two approaches for MSRS were discussed such as global MSRS and local MSRS. These two approaches differ from each other in generation of random points and also in the selection of weights for the distance and response surface criterion. In the global MSRS the set of candidate points are randomly generated. The weight factor is allowed to cycle from high to low for each of the criterion in order to establish a balance. In the case of local MSRS, candidate points are

generated by adding random perturbations to the best solution and providing high weight to the distance criterion so as to find the candidate points close to the current best solution. Another method called multistart local MSRS was proposed, in which the algorithm restarts if it failed to produce better function value after certain iterations. The process restarts from the beginning by defining a new set of initial points. Regis and Shoemaker [61] proposed a method called constrained optimization using response surfaces in which the point selection is achieved by specifying constraints. They used radial basis functions as the metamodel and were limited to continuous functions. Another study by Regis and Shoemaker [63] proposed an algorithm in which they parallelized two different radial basis functions from other studies. The objective of this study was to identify multiple points for optimization. Regis and Shoemaker in 2013 [64]proposed a multistart approach to surrogate optimization that searches for the local minimum of the objective function. The experimental studies were conducted only using radial basis functions, and it is unclear whether the method is metamodel sensitive.

Booker et.al [8] developed a method to optimize some engineering design problems using approximation. This method generates a sequence of approximations for the objective function, which are then used as the surrogates for optimization. In 2000, Neelakantan and Pundarikanthan [55] studied the complex characteristics of a reservoir system using a simulation model. A neural network model is developed to simulate the operation of the reservoir system, which is then optimized using a nonlinear programming optimization model. A simulation model is developed in Azadivar and Tompkins [5] for qualitative decision variables in which simulation models are automatically generated for the given combination of decision policies (categorical variables). The response generated by the simulation model is also qualitative. The variables for each simulation are determined by an object-oriented selection process, and the best solution is obtained

using a genetic algorithm. Queipo et.al. in 2005 [59] developed a surrogate for high-fidelity models that are computationally time consuming and expensive. The surrogates are constructed using data drawn from high-fidelity models and provide fast approximations. The application discussed is the design of liquid rocket injectors.

Giunta, et. al. [29] developed a response surface modeling method to enable aircraft multidisciplinary design optimization. Giunta, et. al. [29] also compared two different approximation methods; quadratic polynomial models and kriging with one, five, and ten independent variables. Jones et.al. in 1998 [21] used kriging as the surrogate model to develop an efficient global optimization of expensive black box functions. This model is more applicable for nonlinear multimodal functions. Jin, et.al. [40] compared four different surrogate models, polynomial regression, multivariate adaptive regression splines, radial basis functions, and kriging. Performance of each model was evaluated based on accuracy, robustness, efficiency, transparency, and conceptual simplicity.

Hosder et.al. in 2001 [66] implemented polynomial response surfaces for multidisciplinary optimization of aircraft with 30 decision variables. Jones [20] discussed seven different methods to represent a taxonomy of response surface based global optimization. A global optimization method based on a general response surface technique is introduced by Gutmann [32]. Bjorkman and Holmstorm [48] developed an improved radial basis function algorithm based on Gutmann (2001) with some extensions. They used metamodels based on kriging techniques in strategy based algorithms, which were developed by Emmerich et.al. [49]. Some issues that occur in surrogate based analysis and optimization were discussed in Queipo, et al. [59]. Wang and Shan in 2007 [27] presented a review of different metamodeling methods and their role in design optimization. A global optimization procedure, by combining multivariate adaptive regression splines and response surface methodology, was presented in Crino

and Brown 2007 [65]. Some information about surrogate modeling methods and their use in optimization strategies were discussed in Forrester and Keane [25].

Simpson et. al. [40] studied various metamodels for design and analysis of computer experiments (DACE) through statistical modeling and optimization. Different metamodels like kriging, radial basis functions, neural networks, and response surface were considered for the study. The main approach was to use all methods and to break down a complex problem, which helps to build a metamodel efficiently. Better search in the design space is also achieved through optimization. Leary et. al. [74] proposed a derivative based surrogate model for approximation and optimization of a computer model. Two different optimization techniques are used, direct optimization and expected improvement optimization. A metamodel was developed using kriging.

Azadivar [4] discussed some issues on simulation optimization. Some of the issues studied were non-existence of analytical expression for objective functions, objective functions that are stochastic for a given deterministic simulation, computer models that are expensive to run, and how the design/modeling language is totally different for simulation and optimization. Carson and Maria [12] reviewed some methods employed for simulation optimization and some applications. Some of the simulation optimization methods discussed are gradient based search methods like finite differences, likelihood ratios, perturbation analysis, frequency domain methods, stochastic optimization, response surface methodology, heuristic methods like genetic algorithms, evolutionary strategies, simulated annealing, Tabu search, and Nelder and Mead's simplex search. Finally some statistical methods are reviewed, like importance sampling methods, ranking selection, and multiple comparisons.

Swisher et.al. in 2003 [76] compared two frequently used statistical methods, ranking and selection, and multiple comparison procedures for selecting the best design

through simulation optimization. Truong and Azadivar [79] proposed a hybrid optimization approach to optimize the supply chain configuration problem. Simulation, mixed integer programming, and genetic algorithms are used to create this hybrid model. The objective of the model is to increase the performance of the supply chain. A supply chain simulator is used to create the metamodel, which is then optimized.

A sequential based kriging optimization, also called efficient global optimization, is explained in Huang et.al. [38]. This is widely used in circuit board manufacturing and other manufacturing applications. In this method, an initial metamodel is built, and kriging is used to predict the responses. Then a cross validation approach is used to check if the predicted responses are valid. The location of the new evaluation points are determined to check if there is any improvement in the metamodel, and the metamodel is updated accordingly with the new data points. Optimization of a Latin-hypercube design (LHD) using sequential sampling is discussed in Xiong et. al. [82]. An efficient quasi-random sequential sampling method is described in which the sequential sampling is formulated as an optimization problem. Maximin LHD is used to determine the design points.

Williams et al. in 2000 [81] proposed a sequential design approach to find an optimal response by taking into account both the controllable variables and uncontrollable variables that affected the response function. The objective function depends only on the controllable variables. Weighted averages of responses are minimized over the values of environmental variables. A comparison of several regression models to determine an optimal design is explained in Atkinson and Fedorov [3]. Chernoff, [15] developed a procedure for sequential design of experiments. In his method, based on each selected observation, a decision is made whether to proceed with computer experiments. Several experimental techniques that maximize the information per unit are described. The study is motivated by testing a simple null hypothesis versus a simple alternative hypothesis for

a given computer experiment. Lei et. al. [47] compared three different sequential optimization methods, namely a sequential least squares method, a sequential kriging method, and a sequential linear Bayesian method. These techniques were implemented on the optimization design for an electromagnetic device. Design point improvisation is done by following a special technique called coarse optimization in a loop. Results showed that this updating method reduced the number of sample points significantly. Metamodels are created using kriging and linear Bayesian estimation.

Improving the efficiency of DACE using a spatial correlation metamodel of computer based engineering analyses is discussed in Simpson et. al. [70]. Three different case studies were used to validate the metamodel. Gramacy and Lee [30] proposed a model that automatically explores the space while simultaneously fitting the response surface, using predictive uncertainty to guide subsequent experimental runs. A Bayesian treed Gaussian process is used as the surrogate model. An adaptive sequential design framework using a hybrid approach is developed that blends optimality from statistics and flexible strategies from active learning in which the learning algorithm has some control over the inputs.

2.1.2 Exploration methods

Exploration is a technique that identifies and explores the key regions in the input space that have not been explored in the past. It also helps to sample points near discontinuities to make sure they are discontinuous but not steep slopes (Crombecq et al., [17]). Exploration methods fill the input space more evenly compared to exploitation. It does not typically need the response to sample the points since the goal is to appropriately sample the input space.

Provost et.al. in 1999 [58] described a progressive exploration sampling method. Two different progressive sampling methods are discussed, arithmetic and geometric.

The objective of using these methods is to arrive at convergence. The arithmetic sampling follows an arithmetic progression in which the difference between consecutive sample points is a constant:

$$S_a = n_0 + (i * n_\Delta) = \{n_0, n_0 + n_\Delta, n_0 + 2n_\Delta, \dots n_0 + kn_\Delta\}.$$

The geometric sampling follows a geometric progression

$$S_g = a^i n_0 = \{n_0, an_0, a^2 n_0, \dots . . a^k n_0\},$$

where S is the sampling schedule, and n_i is the size of the sample. An optimal number of samples are obtained when convergence is detected in both the progressive sampling method. The convergence criteria in the study was based on model quality. Gehrke et.al. [28] proposed a tree-based method of sampling called Bootstrapped Optimistic Algorithm for Tree (BOAT) Construction. This method can handle both numerical and categorical types of data points. The split is based on a technique called coarse splitting.

A hybrid sequential design method of integrating exploration and exploitation is discussed in Crombecq et. al. [17]. This method integrates both these techniques to determine an surrogate approximation for a simulation model. Two different criteria are defined for each technique. Exploration uses a Voronoi approximation, and exploitation uses a local linear approximation. The exploration part in the hybrid design is used to identify an area in the input space. The major goal of the exploitation in the hybrid design is to use the responses from old samples to guide the new data points to interesting areas in the input space.

2.1.2.1 Orthogonal arrays

Orthogonal arrays (Hedayat et.al. [36)] are fractional factorial designs with each factor having multiple levels, such that the full factorial of a subspace is fully represented. For example, a strength two orthogonal array has the full factorial for every two-

dimensional subspace represented. The orthogonal arrays will reduce the input points considerably (Simpson et.al. [72]). The design is orthogonal if the cross-product for each pair of factors (x_i and x_j) of *N* input points is zero, given by:

$$\sum_{u=1}^{N} x_{iu} x_{ju} = 0$$

2.1.2.2 Latin Hypercube Design

The Latin Hypercube Design (LHD) was first used in McKay et al. [51] and was shown to be better than the random sampling (or Monte Carlo) method. LHD ensures that the range of each dimension is well-represented, but LHDs are not typically orthogonal. The range of each input is divided into equal intervals and one observation in each interval is obtained through random sampling. Mathematically it is represented as,

$$x_{ij} = F^{-1}\left[\frac{1}{n}\left(p_{ij} - r_{ij}\right)\right],$$

where x_{ij} is the sample point (i^{th} value of the j^{th} variable), F is a function of the input variables, n is the number of intervals, p_{ij} is the element of an $i \times j$ matrix, and r_{ij} is a sampled value of a uniform $[0,1]$ random variable, which is independent of p.

2.1.2.3 Sobol' Sequence

A Sobol' sequence is a quasi-random low discrepancy sequence introduced by Sobol′ in 1967 [75]. Other low discrepancy sequences are a Faure sequence (Faure [37]), a Halton sequence (Halton [33]), and a Hammersley sequence (Hastie et.al., [35]). These sequences place points more uniformly than the Monte Carlo sampling method. Low discrepancy sampling is frequently applied in numerical integration (Niederreiter [56]). Kung in 2012 [46] proposed a new method of sampling that combines a mixed array design and the Sobol' sequence, which makes it easier to handle both categorical and continuous variables simultaneously.

2.1.2.4 Maximin distance method

Johnson, et. al. [42] proposed a method of sampling for computer experiments called the maximin distance sampling method. In this method the distance between the sample points in the existing set and the new set is determined. The winning candidate points (w) are the ones that maximize the minimum distance between the two sets. It is given by:

$$w \in \arg \max_{y \in x_p} \min_{x \in x_n} \|x - y\|,$$

where $x \in x_n$ and $y \in x_p$

2.1.3 Exploitation Method

Exploitation is a technique of sequential design of experiments that looks for sample points in areas that are already identified as interesting, where the (potential) optima might be. We seek to learn more detail about the region by collecting more points. One method of exploitation is explained in Farhang-Mehr and Azarm [24] in which a Bayesian metamodel approach is used to design the computer experiment in a sequential manner. This approach identifies irregularities in design areas by creating several metamodels using information previously gathered. Another novel method to select points for computer experiments is proposed in Kleijnen and Van [44]. This is an application-driven method since the input/output function is specific for a given simulation, deterministic or stochastic. In this method, a set of possible points is selected using a space-filling design, and the winning candidate is then decided by cross-validation. These new points are added to the original set and then simulated, thereby using fewer points in the actual computer experiment. An adaptive sequential sampling methodology for a metamodel defined with Non Uniform Rational B-splines (NURBs) is proposed in Turner et.al [80]. The initial sampling is done using a traditional factorial

design or Latin hypercube design. However, the sequential sampling is performed by an adaptive method in which the computer design itself is modified between the iterations. The metamodel used in this design is the hyper-dimensional performance model or HyPerModel. This method also employs various nonlinear optimization techniques to determine the new sample points. The method is demonstrated through five different applications that prove its efficacy. Jin et.al. [41] propose a sampling technique using sequential sampling to build a metamodel for optimization of complex systems that includes expensive computer experiments. The sequential sampling is performed by two techniques, namely maximin scaled design approach and cross-validation. The maximin scaled approach is a modification of the traditional maximin distance method, except that a weight parameter is added to the equation. The weight parameter depends on the importance of the variable. The weight equation is given by

$$\text{Max } [\min(d(X_C, X_A))],$$

where $d(v,w) = \sqrt{\sum_{h=1}^{k} a_h (v_h - w_h)^2}$, X_C is the new set of points, X_A is the initial set of points, a_h is the weight parameter, v_h is the point from new set and w_h is the point from the initial set. The types of metamodeling used in this paper are a kriging and a radial basis functions. Two metamodels are used to generate different functions and perform a comparison study. Osio and Amon [57] propose an optimal sampling technique, in which a Bayesian procedure is applied to generate the surrogate model, which is sequentially updated by adding samples to improve accuracy. The samples are selected using a nonlinear programming technique that is solved using a generalized reduced gradient. This methodology is called sampling site optimization.

2.2.2.1 Adaptive sampling

In adaptive sampling, more sample points are collected only on the variables that are of more interest, or variables that highly affect the response (Thompson and Collins, [77]; Thompson and Seber,[78]). This method is commonly used when the population of interest is rare or hard to reach. It also helps to explore hidden points. This method is commonly used in health care problems. A major drawback of this method is that a lot of areas might be left unexplored.

2.2.2.2 Entropy

Entropy was used as a measure of information available by Shannon [69]. Later, Currin et.al [18] modified Shannon's work and used entropy to sample points in design of experiments more like D-optimality, where the covariance matrix is maximized while adding more points (Shannon [69]). Categorical variables cannot be sampled using this method. Entropy sampling is a method to maximize the information available in the sample set and hence the criterion to select the new points is given by (Koehler and Owen [45]):

$$\max(|V_D| * |H^T V_D^{-1} H|),$$

where $V_D$is the correlation matrix of all the points (new, *n* plus existing sample points, *m*), H is the $(n+m)$ vector of $1's$.

2.2.2.3 MSE method sampling method

Mean squared error (MSE) sampling is the method of selecting a new sample point x with the largest prediction error in the existing metamodel. The MSE criterion is given by:

$$w = \max\left(s^2(x)\right),$$

where s^2 is the prediction error of the metamodel and w is the new point.

2.2.2.4 IMSE sampling method

Sacks et.al. [67] proposed the integrated MSE (IMSE) method of sample selection. Let x_n be the set of existing sample points and x_p be the set of new possible sample points, then the selected new set of sample points (x) consists of those that minimize the integrated mean square error. The criteria to select points using the IMSE method is given by:

$$w = \min_x \int s^2(x)dx.$$

2.2.2.5 Cross-validation

Kleijnen and Van [44] used cross-validation for sampling data points in their research. In this method new points are obtained by predicting the output without executing the computer model at each new candidate point and obtaining the variance of the predicted output. This is performed by leaving out one point at a time and determining the prediction error on those left out points. No new sample points are needed in this method. This is given by:

$$e(x) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_{-1}(x) - \hat{y}(x))^2},$$

where n is the total number of points in the existing sample set, $\hat{y}(x)$ is the prediction of the responses of all *n* points and $\hat{y}_{-1}$ is the prediction of the responses after removing one point $(n-1)$.

2.3 Surrogate Modeling or Metamodeling Techniques

The metamodels or the surrogate models are the approximate models that try to replicate the outputs of the simulation that are being studied. A good metamodel can be obtained when a reasonable number of sample points are selected using a sequential design process. Approximation models are often referred to as metamodels because they provide a “model of the model” (Kleijnen [43]). A metamodel can replace an

computationally expensive simulation and hence it is widely used in many engineering applications.

Metamodeling techniques include Response Surface Methodology (Box and Draper [9]; Myers and Montgomery [54]), Artificial Neural Networks (ANN) (Smith [73]; Cheng and Titterington [14]), Kriging methods (Sacks, et al. [67]; Booker, et al. [8]]), Multivariate Adaptive Regression Splines (MARS) (Friedman [26]), Radial Basis Functions (Hardy [34]; Dyn, et al. [22]) Tree models and Treed regression (Alexander and Grimshaw [1]), TreeMARS (Sahu [68]), Polynomial Regression (Simpson et.al. [71]) and linear regression.

Selection of a metamodel depends on several factors like the data sampling technique, nonlinearity of the model, dimensionality, model parameter settings, and variable types used in the design (Jin et.al. [40]). It is important to consider multiple criteria while selecting the metamodel. Plenty of research has studied metamodels in the past, of which a few of them are reviewed below.

2.3.1 Polynomial Function

In a polynomial function, the relationship between the decision variable and the response is modeled as an n^{th} order polynomial.

A second order polynomial function is given by:

$$\hat{y} = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_i \sum_j \beta_{ij} x_i x_j.$$

The noisy functions converge faster when using polynomial regression because of its smoothing capability (Jin et al.[40]).This model can handle larger dimensional problems.

2.3.2 Kriging

A kriging model (Sacks, et al. [67]; Booker, et al. [8]) is a technique to estimate points randomly in the unexplored region of the given samples. The deterministic output of the kriging process is usually treated stochastically.

$$\hat{y} = \sum_{j=1}^{k} \beta_j f_j(x) + Z(x),$$

where $Z(x)$ is stochastic process realization. A major disadvantage of the kriging model is that the fitting process is time consuming.

2.3.3 Multivariate Adaptive Regression Splines

MARS, proposed by Friedman [26], adaptively selects a set of basis functions for approximating the response function using a forward algorithm and a backward algorithm. MARS is written as a statistical linear model form:

$$\hat{y} = \beta_0 + \sum_{m=1}^{M} \beta_m B_m(x),$$

where for each $m = 1, \dots, M$, B_m is a basis function and β_m is its modelcoefficient. The basis function of the MARS model consists of a single or a product of truncated linear models. A single truncated function is given by

$$(x-t)_+ = \begin{cases} x - t & x > t \\ 0 \text{ otherwise} \end{cases}$$

or

$$(t-x)_+ = \begin{cases} t - x & x < t \\ 0 \text{ otherwise} \end{cases}$$

where the functions $(x-t)_+$ and $(t-x)_+$ are called as the reflected pair, and *t* is the knot of the basis function. The basis function is given by:

$$B_m(x) = \prod_{k=1}^{K_m} \left[S_{k,m} \left(x_{v(k,m)} - t_{k,m} \right) \right]_+$$

where K_m is the number of factors in the m^{th} basis function, $S_{k,m}$ = +/-1, $x_{v(k,m)}$ is the v^{th}input variable corresponding to the k^{th} linear function in the m^{th} basis function, and $t_{k,m}$ is a knot location on each of the corresponding variables. The MARS function is validated using a generalized cross-validation (GCV) criterion that is given by:

$$LOF\left(\hat{f}_M\right) = GCV(M) = \frac{\frac{1}{N}\sum_{i=1}^{N}\left[y_i - \hat{f}_M(x_i)\right]^2}{\left[1 - \frac{C(M)}{N}\right]^2}$$

where $C(M)$ is the complex cost function. MARS is not effective in modeling categorical decision variables (Jin et al., 2001).

2.3.4 Radial basis functions

Radial basis functions (Hardy [34]; Dyn, et al. [22]) are helpful for models needing multivariate data interpolation. Radial basis functions use linear combinations of radially symmetric functions like Gaussian, polyharmonic or multiquadric based on some distance metrics like Euclidean distance to approximate the response functions. The form of radial basis functions is given by:

$$\hat{y} = \sum_{i=1}^{p} a_i \|x - x_{oi}\|,$$

where $\hat{y}$, approximate function is given by the sum of p radial basis functions with different center x_{oi} and a coefficient a_i.Radial basis function is suitable for deterministic computer experiments.

Jin et.al [40] did a comparison study between four different metamodels namely, polynomial regression, kriging, MARS, and radial basis functions.

2.3.5 Tree based Models

2.3.5.1 Classification and Regression Tree (CART) Models

Classification and regression trees were developed by Breiman, Freidman, Olshen and Stone (Brieman et al. [10]) to construct decision trees by recursively splitting the input variable space. When the output variable is binary, then the goal is classification, and classification trees are employed. When the output variable is numerical, then the goal is regression, and regression trees are employed. As a result of data splitting, hyper-rectangles are formed at each terminal node over which a simple model is built. Input variables can be both categorical and numerical.

2.3.5.2 Treed regression

Treed regression is a technique proposed by Alexander and Grimshaw [1] that modifies the regression tree structure to fit linear regression models at the terminal nodes. The tree part of the model can handle categorical and numerical variables, whereas the terminal nodes regression models consist of only the numerical variables.

2.3.5.3 TreeMARS

Sahu [68] developed a new tree based model in which MARS is fit at the terminal nodes, instead of linear regression as in the case of Treed regression. MARS functions handle only the numerical variables, whereas the tree handles both the categorical and numerical variables. If the underlying function is complex, TreeMARS can yield a better approximate model than treed regression.

2.4 Contribution

In this dissertation, an approach for surrogate optimization is presented that combines exploration and exploitation in a process that seeks to find an optimum faster than the methods available in literature. Most of the surrogate-based optimization

methods available in literature focus on augmenting the surrogate optimization or the surrogate models and weight factors (Müller and Shoemaker [52]; Regis and Shoemaker [62]). The proposed method combines the notions of exploration and exploitation without using a weight factor, which has not been previously studied. This method is not concerned about the type of metamodel used, but rather addresses how to search the input space for an optimum using both exploration and exploitation criteria. Most of the existing surrogate optimization methods (Regis and Shoemaker [62]) use multiple criteria to locate points, with a weight factor added to each criterion. Hence the limitation of these methods is to determine the weight factor before optimization or during optimization. The proposed method uses two different criteria to select points using a Pareto (Censor,1977 [15]) approach without any weightage factor added to the criterion.

Chapter 3

Exploration and Exploitation Pareto Approach (EEPA)

In this research a unique approach called Exploration and Exploitation Pareto Approach (EEPA) is developed to combine the notions of exploration and exploitation in the optimization part of the surrogate optimization algorithm. Classic surrogate optimization is performed by starting with sampling points using a space filling design over the input space, collecting the response for each of the sample points typically using computer experiments, constructing a surrogate model, and then finally performing the optimization using the surrogates. Surrogate optimization methods discussed in Chapter 2, focus more on metamodel selection, choosing points randomly to optimize the metamodel. In this research, the surrogate optimization is achieved by adding point(s) sequentially by simultaneously using exploration and exploitation techniques.

3.1 Exploration and Exploitation Pareto Approach (EEPA)

Exploration looks for point(s) in areas that have not been explored, whereas exploitation looks for point(s) in areas that have already been explored and found to be promising for optimal solution. EEPA combines both the ideas of exploration and exploitation in a novel approach for surrogate optimization. EEPA process uses two types of metrics, specifically, representation of the input space and the predicted response. The representation of the input space metric is the exploration part that ensures the new points sample unexplored areas. The predicted response metric is the exploitation part that identifies regions where the optimal solution seems promising. A detailed explanation of EEPA is given below, specifically using the maximin distance exploration metric (refer to Section 2.1.2.5) The computational study settings are explained in Chapter 4, and the results are discussed in Chapter 5.

Algorithm 3-1 EEPA Algorithm

A is the initial set, R is the random data set

For each $x \in A$, determine $f(x)$

While $k < K$

Construct a metamodel $\hat{f}$ using $\{(x, f(x)) \mid x \in A\}$

For each $x \in R$, determine $\hat{f}(x)$

For each $x \in R$, determine $d(x) = \min_{\tilde{x} \in A} ||x - \tilde{x}||$

Determine $F = \{x \in R \mid \nexists\, \tilde{x} \in R, \hat{f}(\tilde{x}) \leq \hat{f}(x)$ and $d(\tilde{x}) \geq d(x)\}$

Set $F' = \emptyset$; $k' = 1$

Determine $F' = F' \cup \{x\}, x \in \arg\min \{\hat{f}(x) \mid x \in F\}$

While $k' < K'$ and $F' \subset F$

For each $x \in F$, determine $d'(x) = \min_{\tilde{x} \in A \cup F'} \|x - \tilde{x}\|$

Determine $F' = F' \cup \{x\}, x \in \arg\max \{d'(x) \mid x \in F'\}$

$k' = k' + 1$

End while

For each $x \in F'$, determine $f(x)$

$A = A \cup F'$

$R = R \backslash \{F'\}$

End while

3.3.1 Step 1 – Initialization

EEPA is initialized by creating an initial data set A with i observations and j decision variables or factors. A random data set R is also generated referred to as “weak” points from which candidate points will be selected. The Sobol′ sequence is used to

sample points in A whereas Monte Carlo sampling method is used to sample points in R. The size of A is small whereas size of R is large.

3.3.1.1 Handling Categorical Variables

A challenging part of this technique is sampling categorical variables and discrete-numerical variables alongside continuous variables. In 2012, Kung [46] used different sampling techniques to sample continuous variables and categorical variables separately, and then combined them. In this research all variables are sampled as continuous variables including the categorical variables. The categorical variables are scaled to continuous space with a range of [-1,1] using a uniform distribution. Once the categorical variables are converted to continuous variables, they are combined with the other continuous variables and sampled together using the Sobol' sequence. They are scaled back to the categorical space using the rounding method from Martinez [50]

If the variable has two factor levels the sign on the sampled value will decide which factor level should be selected for the experimental run. Let a and b be the two factor levels of variable x. If the sampled value is positive, the variable takes the value a. If it is negative, then the variable takes the value b.

if $x \geq 0$, then x=factor level a

if $x < 0$, then x= factor level b

If the variable has more than two factor levels, a threshold value, τ as given by Martinez [50] is used to decide which factor level of the categorical variable will be used for the experimental run.

$$\tau = \frac{2}{n - \sqrt[1]{n}} - 1,$$

where n is the number of levels. Only $n - 1$ levels are scaled to the continuous space in the Sobol' sequence, leaving out one of the factor levels of the variable, usually the last

level. The maximum of the values corresponding to $n-1$ levels of each variable is compared with the τ value. If the maximum value is greater than τ, then the factor level corresponding to the maximum value is selected for that experimental run. If τ is greater than the maximum value, then the left out variable is selected for that experimental run. An example of a variable with 4 levels is shown below.

In this example, x_1 has four factor levels, namely a, b, c or d, and the left out factor level is d. Let x_{1a}, x_{1b} and x_{1c} be the values of factor levels a, b and c, respectively, in the continuous space.

$x_1 = a, b, c$ or d, factor level "d" is left out

After sampling in continuous space [-1,1]

let $x_{1a} < 0$

$x_{1b} > 0$

$x_{1c} < 0$

τ = 0.2599 (using n=4 in τ equation).

Now, the max$\{x_{1a}, x_{1b}, x_{1c}\} = x_{1b}$.

Comparing with τ, if $x_{1b} > \tau$, then the selected factor level $= x_{1b}$, else the selected level $= x_{1d}$.

Thus, the categorical variables are sampled along with the continuous variables in the continuous space using Sobol' sequence

3.3.2 Step 2 – Computer Experiment

Once the initial points are obtained in the initial training data set A, the next step is to run the input points on the computer model to obtain the responses. Computer models are black box functions since the function with which the outputs are generated is unknown. Computer experiments are the time consuming part of EEPA.

3.3.3 Step 3 – Metamodel

The next step is to fit a metamodel to the data set A for further study. Several metamodels can be used, depending on the structure of the data, as discussed in Section 2.3. Three different metmaodels are used in this research, Multivariate Adaptive Regression Splines, Radial Basis Gaussian Function and Treed Regression.

3.3.4 Step 4 – Pareto Frontier

Once the metamodel is obtained, the next step is to determine the two criteria, namely, distance and the predicted response to determine the Pareto optimality. Pareto efficiency or Pareto optimality is the method of making one metric better without compromising other metrics, in other words trade-offs or non-dominance between two or more metrics. Since EEPA uses multiple criteria to select points, a Pareto method is preferred so that importance is given for both the metrics. Pareto method aims to identify the non-dominant solution given the two objectives.

The distance metric is obtained by calculating Euclidean distance d between R and A. The predicted response metric is obtained by calculating the estimated output values for R using the metamodel obtained from A. The objective of the optimization is to select point(s) with minimum response and in unexplored regions that will eventually find the minimum solution. Thus the efficient frontier (set of candidate points) is obtained using the points from R by minimizing the predicted response $(\hat{y})$ and maximizing the distance d. The non-dominance solutions are on the efficient or Pareto frontier F given by the following equation.

$$F = \{x \in R \mid \nexists\, \tilde{x} \in R, \hat{y}_i(\tilde{x}) \leq \hat{y}_i(x) \text{ and } d(\tilde{x}) \geq d(x)\}$$

3.3.5 Step 5 – Candidate points

Once the efficient or Pareto frontier is determined, the next step is to find the winning candidate points from the efficient frontier to add to the initial set A. The points on the efficient frontier are our winning candidates, so we add them to our initial set, A. In some cases, depending on the type of metamodel we use, there could be lot of points on the efficient frontier that are close to each other. In this case, we conduct a search using exploration to find the points in unexplored regions. The maximin exploration technique is used for this search that maximizes the minimum distance between the points. The winning candidate points are evaluated on the computer experiment to obtain the responses.

3.3.6 Step 6 – Check for stopping criteria

Currently, the minimum response found after k evaluations is the best solution. The value of k is fixed at the beginning, but other stopping criteria could be developed. The number of evaluations also includes the points in the initial data set. The value of k is limited to a small number since it is necessary to reduce the number of function evaluations of the computer model.

3.3.7 Step 7 – Update the initial set

If the stopping criteria are not met, the optimization proceeds by adding the winning candidate points to the initial set A and steps 3 to 6 are repeated.

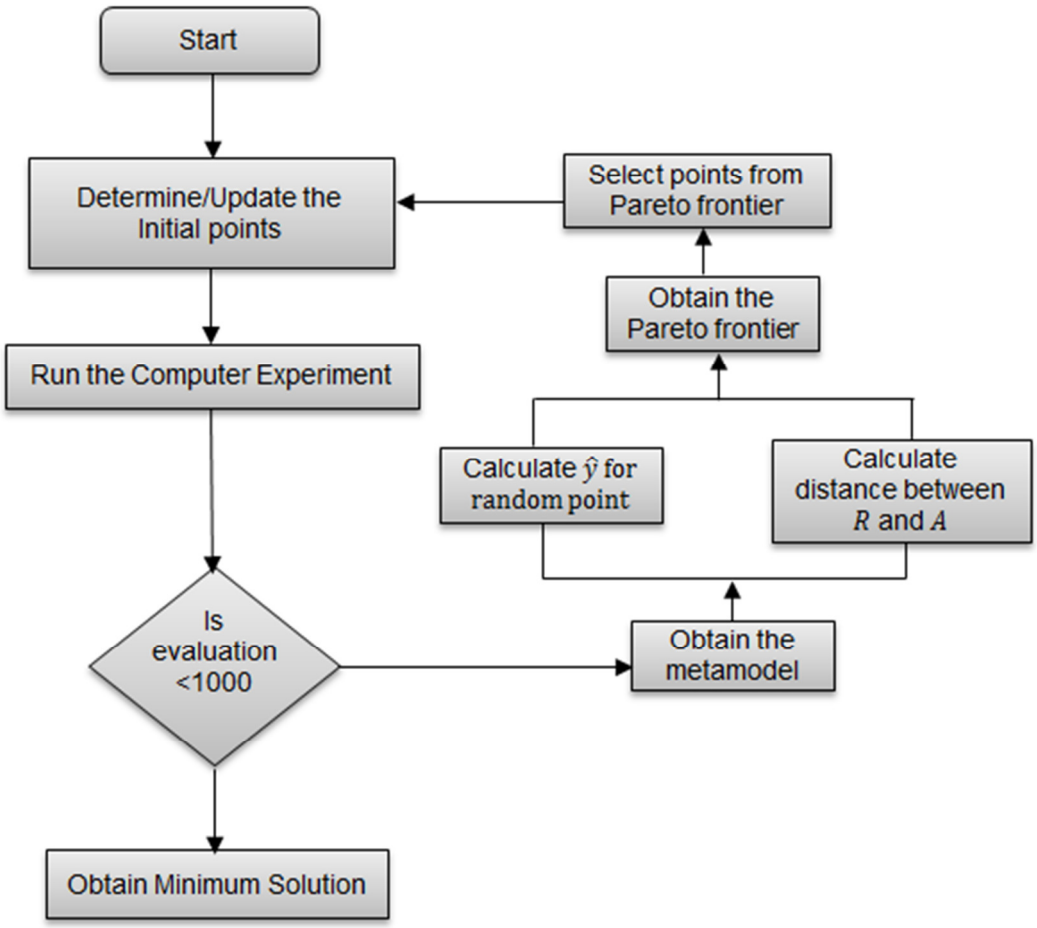

Figure 3-1 EEPA

3.2 Exploration

Exploration is the process of exploring new points in regions of the input space that have not been explored. It uses only the input variables and not the computer model output to determine new points. The process is shown in Figure 3-2. In this research, pure exploration is performed using the methods maximin distance, cosine, Monte Carlo search, and Sobol' sequence. Let A be the initial data set in which the points are generated using the Sobol' sequence and R be the data set that is randomly generated. The objective of the exploration is to select the best points using the above methods from

R and add them to A. The points are added iteratively as shown in the Figure 3-2 until the stopping criterion is met. The stopping criterion for exploration is same as that of EEPA, maximum number of (computer model) function evaluations.

3.2.1 Maximin distance method

The maximin distance method optimizes the surrogate model by finding points in the input space that are in unexplored regions. This could be achieved by calculating the Euclidean distance between vectors. As shown in Figure 3-2, when needed, new points are added by finding the Euclidean distance between every vector in set R and A. The winning candidate is the point that maximizes the minimum distance between the vectors. The process is iterative in which n points are added to A to see if the stopping criteria is met, if not, n more points are added, and the process continues until k evaluations have been performed.

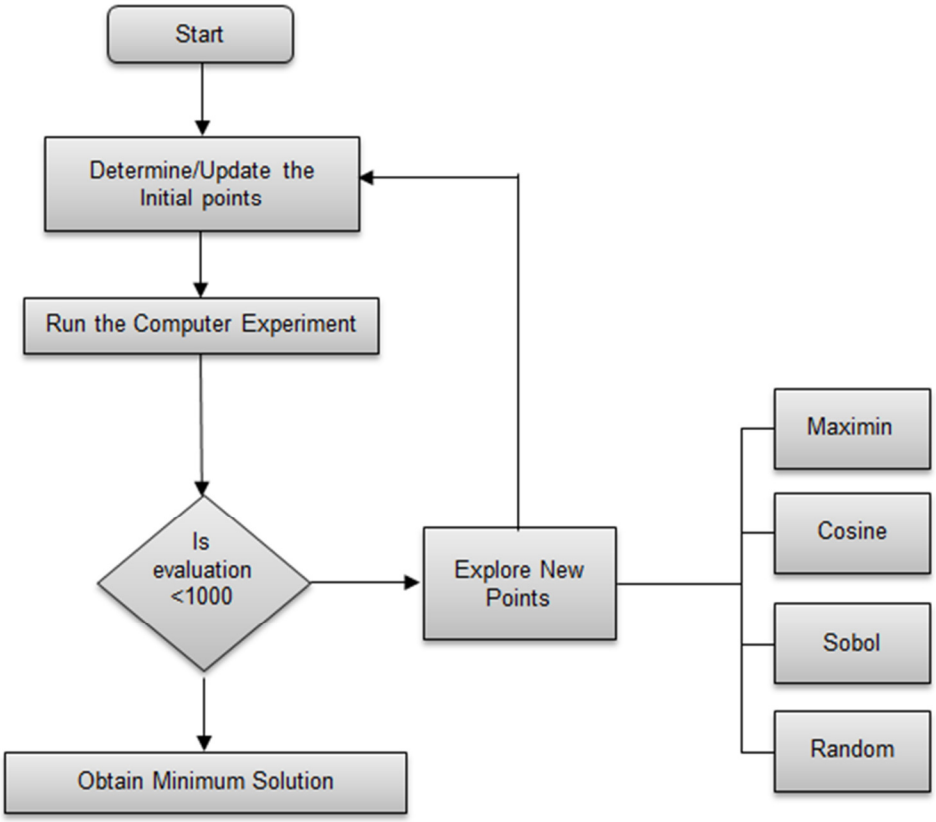

Figure 3-2 Exploration Process

Algorithm 3-2 Maximin distance search method

A is the initial set, R is the random data set

For each $x \in A$, determine $f(x)$

While $k < K$

for $x \in R$

For each $x \in R$, determine $d(x) = \min_{\tilde{x} \in A} \|x - \tilde{x}\|$

Determine $F = F \cup \{x\}, x \in \arg\max \{ d(x) \mid x \in R \}$

$k = k + 1$

For each $x \in F$, determine $f(x)$

$A = A \cup F$

$R = R \backslash \{F\}$

End while

3.1.2 Cosine method

The cosine method is used to find new points in the input space that are in unexplored regions using the angular cosine distance between vectors. The winning candidate is the point that maximizes the minimum cosine distance between the vectors. The process is iterative in which n points are added to A in each iteration until the stopping criterion is met, as discussed in Section 3.1.

Algorithm 3-3 Cosine method

A is the initial set, R is the random data set

For each $x \in A$, determine $f(x)$

While $k < K$

for $x \in R$

For each $x \in R$, determine $d(x) = \min_{\tilde{x} \in A} \frac{\tilde{x} \cdot x}{\|\tilde{x}\|\|x\|}$

Determine $F = F \cup \{x\}, x \in \arg\max \{ d(x) \mid x \in R \}$

$k = k + 1$

For each $x \in F$, determine $f(x)$

$A = A \cup F$

$R = R \backslash \{F\}$

End While

3.1.3 Monte Carlo Search method

The random search method looks for new points randomly in the input space and adds them to the primary data set. In this method, n points are randomly selected from R and added to A. The stopping criterion is checked at the end of each iteration.

3.1.4 Sobol' sequence method

In the Sobol' sequence method, m points are generated using the Sobol' sequence. $1 \ldots m1 \in m$ points in the order in which they appear in the Sobol' sequence are considered as A, while for exploration, n points are added to A in the sequence in the order they appear in m from the remaining $m - m1$ points. Note that $m1$ is same as A in all the other methods. The stopping criterion is checked at the end of each iteration.

Algorithm 3-4 Sobol' Sequence method

M is the Sobol′ set,

$A = 1 \ldots m1 \in M$

For each $x \in A$, determine $f(x)$

While $k < K$ do

$F = A \cup A + 1$

For each $x \in F$, determine $f(x)$

$A = A \cup F$

$k = k + 1$

End while

3.3 Exploitation

The exploitation method is another approach to surrogate-based optimization in which the points are chosen using actual or predicted responses. An exploitation metric can focus on finding at the optimum or seek to minimize the prediction error of the surrogate model. In the current study, the optimum is sought. it samples near the possible discontinuities and also verifies they are actually discontinuous rather than steep slopes [17].

3.2.1 Minimum response method

A surrogate model is built on the initial set A, and the predicted responses are obtained for points on R. In the exploitation process, we add points to set A based on the predicted response. In this process, initially we add the lower corner point as the first point, so that we have the minimum point of the metamodel obtained. After adding the corner point, more point(s) are added that have minimal predicted response. The lower corner point is obtained from the sign of the coefficient of each variable in the surrogate model. For example, in the case of MARS model, if the sign for a coefficient is negative then we select the upper bound or if the sign is positive we select the lower bound for that variable, thereby we add the first point to A. The next step is to find candidates that have minimum responses in R and add them to A. The actual responses are then obtained using the computer model for the winning candidates. The process is repeated until the stopping criterion is met. The minimum solution in A is the known solution.

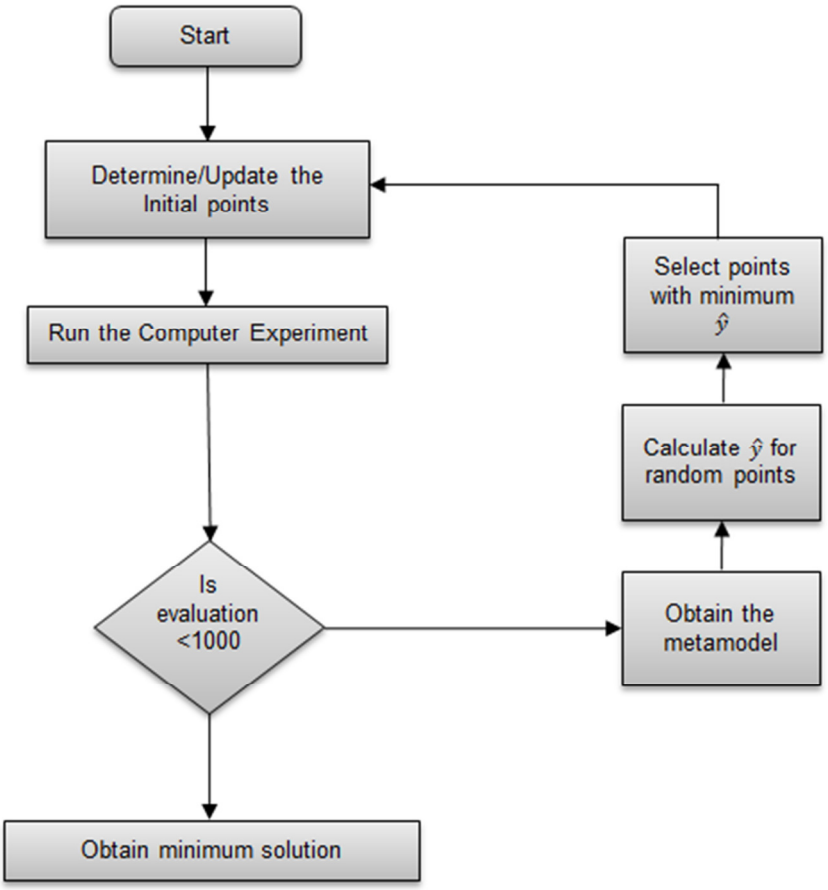

Figure 3-3 Exploitation Process

Chapter 4

Computational Study

EEPA was tested on various test functions and then compared to pure exploration and exploitation methods. Among the exploration methods, maximin, cosine, Monte Carlo sampling and the Sobol' sequence method were used in this study. Minimum response is used as the exploitation method metric. The test functions used in this study have been used in various methods in the literature to test other surrogate-based optimization methods (e.g., Muller and Shoemaker 53). All these test functions except the sphere function are non-convex. Except for green building function all other function has one global minimum.

4.1 Experimental Set-up

4.1.1 Green building function

In this research, a black-boxtest function is also used to test EEPA for which green building case study is considered. The data for the green building case study is obtained from a computer model called eQUEST [23]. eQUEST is a menu driven tool and so the simulation consumes a lot of time. The surrogate optimization using EEPA will search for best possible points to find the minimum solution

4.1.1.2 Green building decision making framework

The green building model is based on the multistage multi objective decision making framework for green building model developed by Kung [43]. The decision variables correspond to the building options, and the multiple responses considered in this study are the operational performance metrics of the building. The data collection part is performed by the computer model called eQUEST.

The decision making framework model has 46 different decision variables. Categorical variables are specified as binaries. The response measured is the annual source energy total in million British thermal units (Mbtu). The case study used is a single story 2500 square foot building. In addition to the 46 controllable decision variables, 68 uncontrollable variables were also considered, which follow uniform distributions.

The input points for the experimental runs in eQUEST were obtained using design of experiments (DOE) techniques. A unique hybrid model by combining mixed array and Sobol' sequence was developed to combine the categorical and quantitative variable in a single run.

Finally the approximate model (metamodel) for the multiple response green building models is developed using tree-based models that use seemingly unrelated regressions. Tree based models called Treed regression is used as the metamodel.

4.1.1.3 Green Building Simulation Tools

Building simulation tools [11] help analyze different aspects of the building. The software determines the performance of the building given the building options. For example, the types of windows, the type of insulation material, and the thickness of concrete. Some of the software available includes eQUEST, ATHENA Impact Estimator for buildings (ATHENA) [2] and Building for Environmental and Economic Sustainability (BEES) [6].

The eQUEST tool, developed by James J. Hirsch & Associates, California for the Department of Energy (DOE), allows detailed analysis of building design technologies. Some of the performance metrics measured in eQUEST is life cycle cost, energy cost, utility cost, and HVAC utilization cost. The eQUEST tool combines a building creation wizard, an energy efficiency measure wizard, and a graphical results display module with a simulation engine to analyze the building design. The eQUEST tool is derived from an

advanced version of the DOE-2 building energy use simulation program. The options for each decision variable are specified in the software. The user can only select one of the options available under each of the decision variable. The eQUEST software has the ability to analyze an entire building and give a response subject to the entire effect of the building. There are 41 different wizards (screens) in the eQUEST software from which the user specifies the building information. Some of the information includes building orientation, landscape, windows, insulation, HVAC system, construction type, and roof type. Some of the information available in different wizards is related to each other. The simulation runs only if all of the information in the 41 wizards is provided. Once the simulation is done, the reports are specified in table format with all the response variables. This tool gives the design or construction engineer more detail about the building.

ATHENA represents the performance of the building in terms of environmental impact. Users can select different building options and how those will affect the environment. ATHENA has also the capability of user defining decision variables and custom variables. ATHENA however does not consider the life cycle cost of the building.

BEES represents the performance of the building in an economic and environmental impact score. BEES does not establish a balance between sustainability and energy efficiency and hence it is not commonly used.

This research uses only eQUEST as the computer model.

4.1.1.4 True function

The true function is obtained from 200 data points using the treed regression model that will be used instead of the eQUEST computer model in this research to obtain the response in EEPA, exploitation and exploration methods. This is to simplify the surrogate optimization process and do more trials.

4.1.2 Levy's Function

Levy's function is given by equation 4-1

$f(x) = sin^2(\pi w_i) + \sum_{i=1}^{d-1}(w_i - 1)^2[1 + 10\ sin^2(\pi w_i + 1)] + (w_d - 1)^2\ [1 + \sin^2(2\pi w_d)]$, (4-1)

where, $w_i = 1 + \frac{x_i - 1}{4}$, for all $i = 1, \dots, d$

Number of dimensions used in Levy's function is 20 and $x_i \in [-10, 10]$, for all $i = 1 \dots 20$.

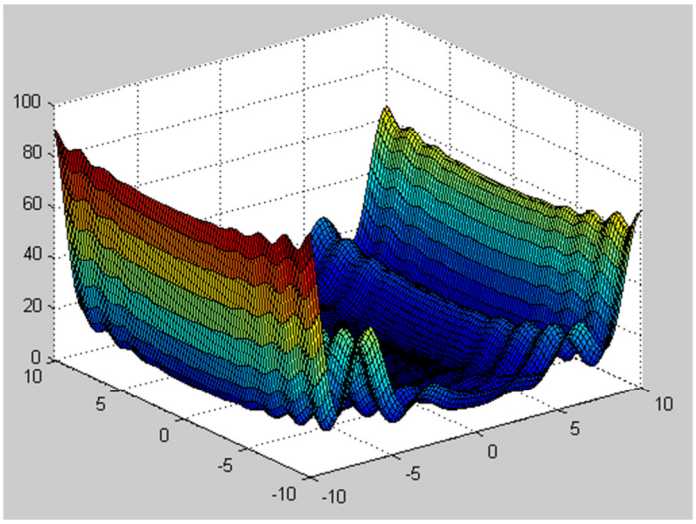

Figure 4-1 Levy Function

4.1.3 Rosenbrock Function

The Rosenbrock function is generalized using equation (4-2)

$f(x) = \sum_{i=1}^{d-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$ (4-2)

Number of dimensions used in Rosenbrock function is 16 and $x_i \in [-6, 4]$, for all $i = 1 \dots 16$

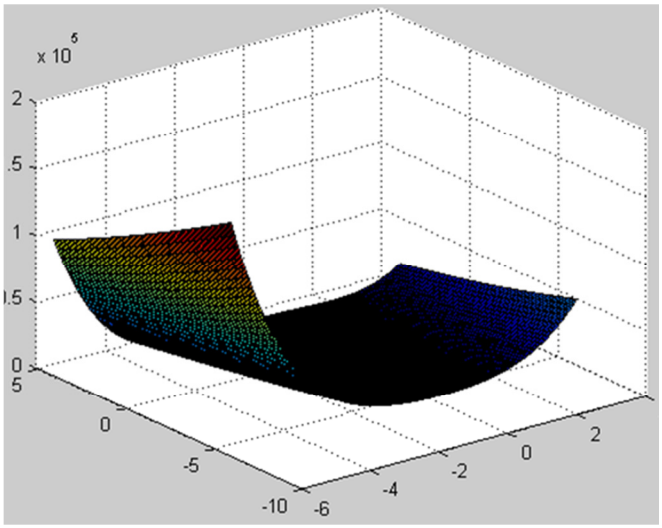

Figure 4-2 Rosenbrock Function

4.1.4 Rastrigin Function

Rastrigin function can generalized in equation 4-3

$$f(x) = 10d + \sum_{i=1}^{d}[x_i^2 - 10\cos(2\pi x_i)] \quad (4\text{-}3)$$

Number of dimensions used for Rastrigin function is 30 and $x_i \in [-5.12, 5.12]$, for all $i = 1 \ldots 30$.

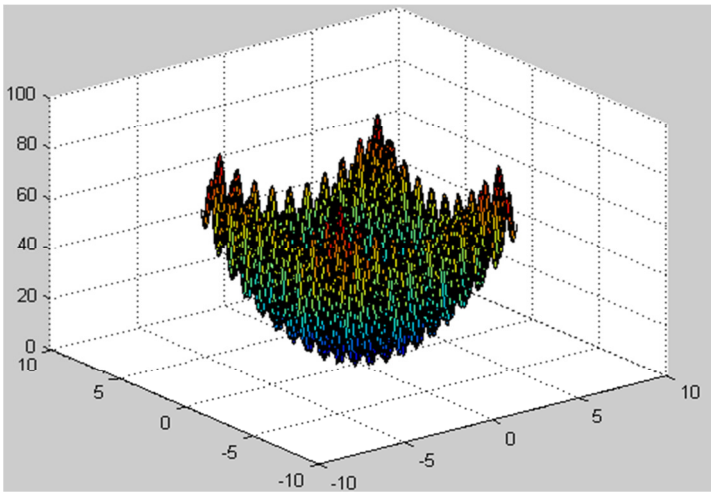

Figure 4-3 Rastrigin Function

4.1.5 Sphere Function

The sphere function is given by the equation 4-4

$$f(x) = \sum_{i=1}^{d} x_i^2 \quad (4\text{-}4)$$

Sphere function is approximated using 27 dimensions and $x_i \in [-5, 5]$, for all $i = 1 \ldots 27$

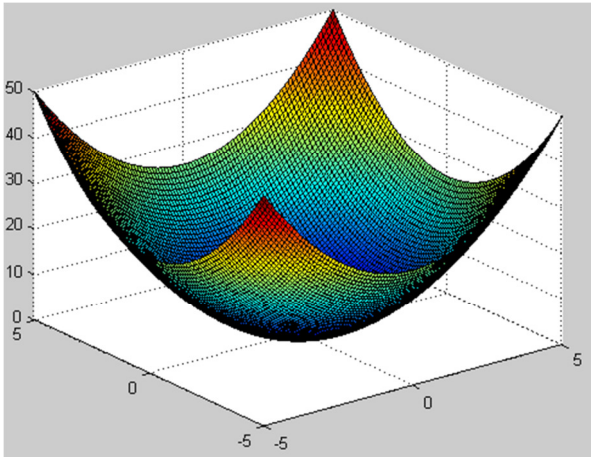

Figure 4-4 Sphere Function

4.2 Metamodel

Two different metamodels were used in this experimental study, Multivariate Adaptive Regression Splines (MARS) and Gaussian Radial Basis Functions (RBF), which are discussed in section 2.3.A metamodel is only used for EEPA and pure exploitation process.

Chapter 5

Optimization of Green Building (GB) BasedTreed Regression Model

5.1 Green Building (GB) BasedTreed Regression Model

An optimization model was developed for the green building model (Kung [46]) using mixed integer linear programming. Three different responses were considered for the optimization, treed regression surrogate model for response 1, 2 and 3 are shown in Figures 5.1, 5.2 and 5.3. The variables are specified as categorical with several attributes or levels, discrete numerical or continuous. All 46 decision variables are shown in Table 5-1

Table 5-1 Decision Variables and Their Levels (Kung [46])

Variable	Variable Factor levels	Variable Type
Ground Floor Construction (x1)	• 2 inch Concrete • 4 inch Concrete • 6 inch Concrete • 8 inch Concrete	Discrete-Numerical
Ground Floor Interior Insulation (x2)	• 1 inch Polystyrene • 1 1/2 inch Polystyrene • 2 inch Polystyrene • 3 inch Polystyrene	Discrete-Numerical
Ground Floor Cap (x3)	• 1.25 inch Lightweight Concrete • 2 inch Lightweight Concrete • 3 inch Lightweight Concrete • 4 inch Lightweight Concrete	Discrete-Numerical
Ground Floor Exterior/Cavity Insulation (x4)	• 1 inch Polystyrene • 2 inch Polystyrene • 3 inch Polystyrene • 4 inch Polystyrene	Discrete-Numerical
Exterior Wall Insulation (x5)	• 1 inch Polystyrene • 1 1/2 inch Polystyrene • 2 inch Polystyrene • 3 inch Polystyrene	Discrete-Numerical
Additional Wall Insulation (x6)	• R-3 Batt • R-7 Batt • R-11 Batt • R-13 Batt	Discrete-Numerical

Table 5-1—*Continued*

%Window-North (x7)	• 10% • 15% • 20% • 25%	Discrete-Numerical
%Window-South (x8)	• 10% • 15% • 20% • 25%	Discrete-Numerical
%Window-East (x9)	• 10% • 15% • 20% • 25%	Discrete-Numerical
%Window-West (x10)	• 10% • 15% • 20% • 25%	Discrete-Numerical
Additional Roof Insulation (x11)	• R-7 Batt • R-19 Batt • R-30 Batt • R-49 Batt	Discrete-Numerical
Ceiling Batt Insulation (x12)	• R-13 Batt • R-19 Batt • R-21 Batt • R-30 Batt	Discrete-Numerical
Exterior Roof Insulation (x13)	• 1 inch Polystyrene • 1 1/2 inch Polystyrene • 2 inch Polystyrene • 3 inch Polystyrene	Discrete-Numerical
Footprint X (x14)	• 100 ft • 70.7 ft • 62.5 ft • 50 ft	Discrete-Numerical
Door Dimension-Width (x15)	• 3 ft • 4 ft • 5 ft • 6 ft	Discrete-Numerical
Door-Frame Width (x16)	• 2 inch • 2.3 inch • 2.7 inch • 3 inch	Discrete-Numerical
Design Max Occupant Density-Residential (General Living Space) (x17)	Range: 575 to 675 (sqft/person)	Continuous

Table 5-1—*Continued*

Design Ventilation-Residential (General Living Space) (x18)	Range: 10 to 30 (CFM/person)	Continuous
Design Max Occupant Density-Residential (Bedroom) (x19)	Range: 575 to 675 (sqft/person)	Continuous
Design Ventilation-Residential (Bedroom) (x20)	Range: 10 to 30 (CFM/person)	Continuous
Design Max Occupant Density- Residential (Garage) (x21)	Range: 575 to 675 (sqft/person)	Continuous
Design Ventilation-Residential (Garage) (x22)	Range: 10 to 30 (CFM/person)	Continuous
Design Max Occupant Density- Dining Area (x23)	Range: 5 to 105 (sqft/person)	Continuous
Design Ventilation-Dining Area (x24)	Range: 10 to 30 (CFM/person)	Continuous
Design Max Occupant Density-Kitchen and Food Preparation (x25)	Range: 250 to 350 (sqft/person)	Continuous
Design Ventilation-Kitchen and Food Preparation (x26)	Range: 5 to 25 (CFM/person)	Continuous
Design Max Occupant Density-Corridor (x27)	Range: 100 to 200 (sqft/person)	Continuous
Design Ventilation-Corridor (x28)	Range: 5 to 25 (CFM/person)	Continuous
Design Max Occupant Density-Laundry (x29)	Range: 100 to 200 (sqft/person)	Continuous
Design Ventilation-Laundry (x30)	Range: 15 to 35 (CFM/person)	Continuous
Design Max Occupant Density-All Others (x31)	Range: 100 to 200 (sqft/person)	Continuous
Design Ventilation-All Others (x32)	Range: 5 to 25 (CFM/person)	Continuous
Wall Construction (x33)	・Wood Frame, 2・4, 16 inch o.c. (a) ・Wood Frame, 2・4, 24 inch o.c. (b)	Discrete-Categorical

Table 5-1—*Continued*

Windows-Glass Category (x34)	• Double Clear/Tint (a) • Double Low-e (e2 = 0.1) (b)	Discrete-Categorical
Roof Construction (x35)	• Wood Advanced Frame, 24 inch o.c. (a) • Wood Advanced Frame, >24 inch o.c. (b)	Discrete-Categorical
Exterior Wall Finishes (x36)	• Brick (a) • Concrete (b)	Discrete-Categorical
Exterior Wall Color (x37)	• Light (a) • Dark (b)	Discrete-Categorical
Interior Wall Insulation (x38)	• None (a) • 1 inch Polystyrene (b)	Discrete-Categorical
Exterior Roof Finish (x39)	• Concrete (a) • Built-up Roof (b)	Discrete-Categorical
Exterior Roof Color (x40)	• Light (a) • Dark (b)	Discrete-Categorical
Doors-Construction (x41)	• Double Clear/Tint (a) • Double Low-e (e2 = 0.1) (b)	Discrete-Categorical
Pitched Roof (x42)	• Without Pitched Roof (a) • With Pitched Roof (b)	Discrete-Categorical
Ceiling Interior Finishes (x43)	• Lay-In Acoustic Tile (a) • Drywall Finish (b) • Plaster Finish (c)	Discrete-Categorical
Windows-Glass Type (x44)	• Clear 1/8, 1/4 inch Air (a) • Clear 1/8, 1/2 inch Air (b) • Clear 1/4, 1/4 inch Air (c) • Clear 1/4, 1/2 inch Air (d)	Discrete-Categorical
Orientation (x45)	• N/S Component (Face North) (a) • N/S Component (Face South) (b) • E/W Component (Face East) (c) • E/W Component (Face West) (d)	Discrete-Categorical
Doors-Glass Type (x46)	• Clear 1/8, 1/4 inch Air (a) • Clear 1/8, 1/2 inch Air (b) • Clear 1/4, 1/4 inch Air (c) • Clear 1/4, 1/2 inch Air (d)	Discrete-Categorical

The split on the tree can be due to categorical variables or quantitative variables as shown in the Figure 5-1, 5-2 and 5-3. The numerical value at the terminal node is the

average of all the responses for observations at that node. The functions f, f', f'' at the terminal nodes represent the linear regression equations, which are functions of variable X_i, with slope coefficient θ_iThe tree is modeled such that each terminal node has at least 35 observations. Response 1 has 8 terminal nodes, response 2 has 7 terminal nodes, and response 3 has 8 terminal nodes. The objective of the proposed model is to detect which is the optimum terminal node.

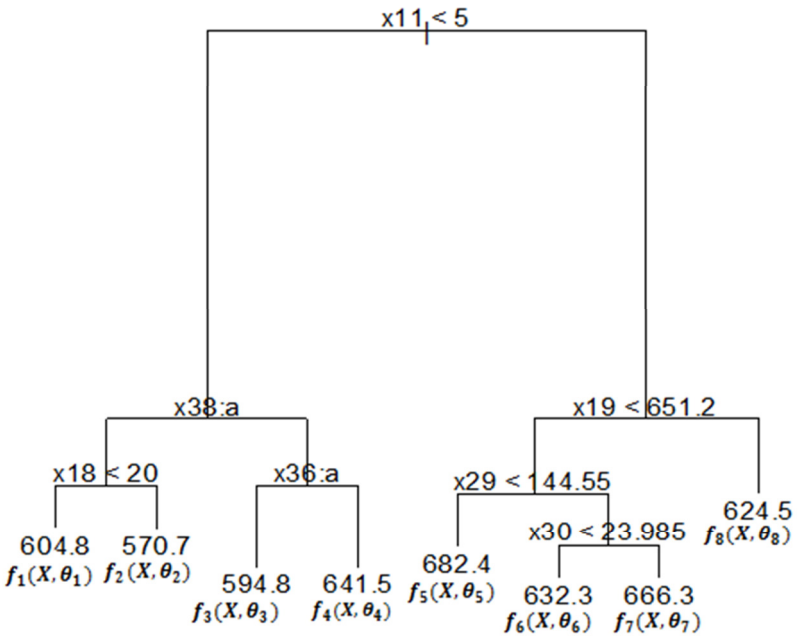

Figure 5-1 Regression tree for response 1

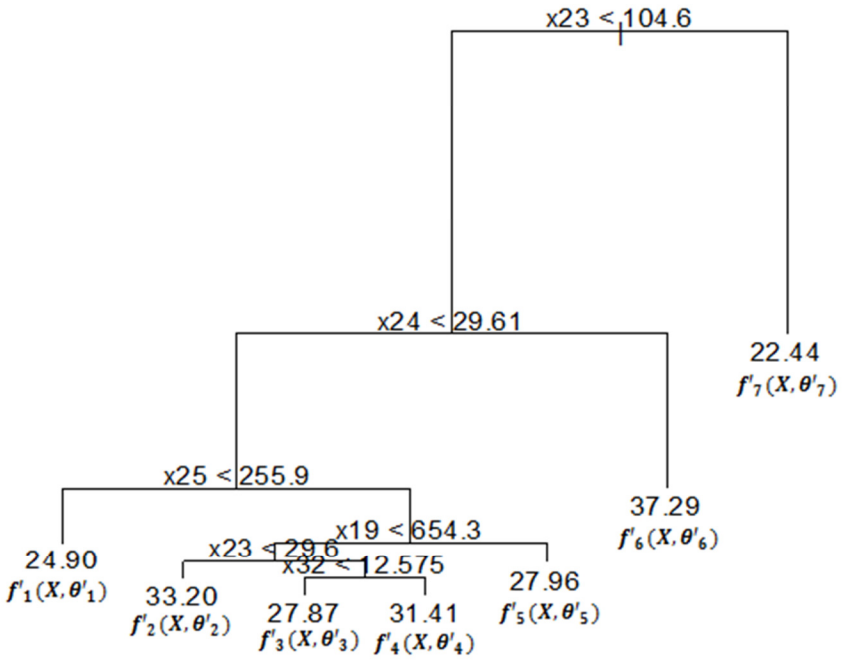

Figure 5-2 Regression tree for response 2

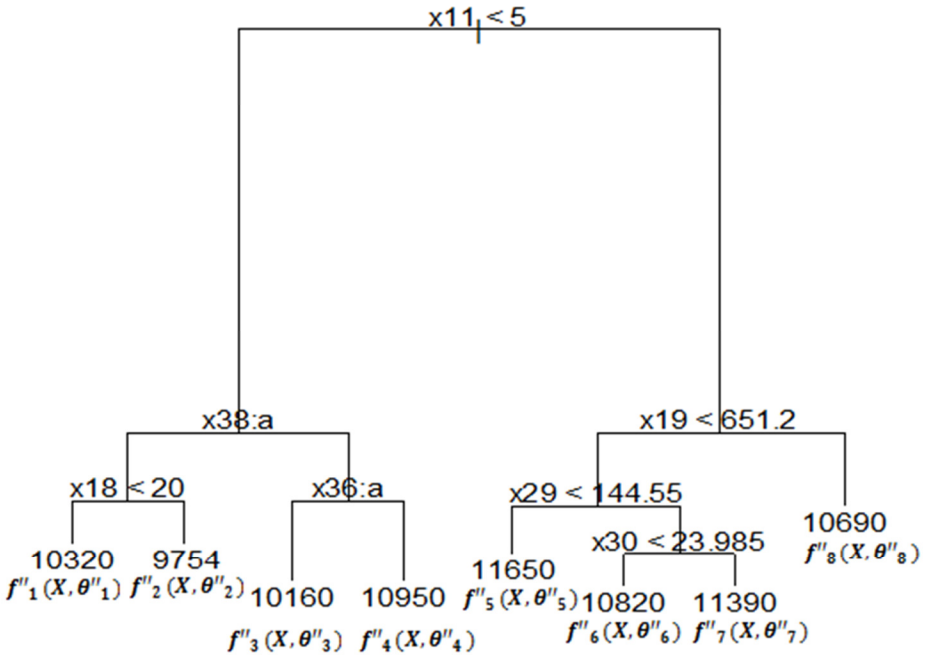

Figure 5-3 Regression tree for response 3

5.1.2 Mixed Integer Linear Programming Model

The objective is a linear function, but the decision variables consist of a mix of real and integer numbers and hence mixed integer linear programming is used for the optimization. The Xpress optimization [31] suite is used to solve the optimization model. Xpress uses one of the four linear programming algorithms, namely primal algorithm, dual algorithm, barrier algorithm or network algorithm (Guéret et al. [31]). In the primal simplex algorithm, each inequality constraint in the linear programming formulation splits the solution space into two halves. Primal simplex looks for the optimal basis while maintaining feasibility. Dual simplex is the opposite for the simplex algorithm and tries to attain primal feasibility while maintaining optimality. In the barrier method, an optimal solution is reached by traversing through the interior of the feasible region (Guéret et al. [31]). In the network algorithm, an optimal solution is attained by traversing through the nodes and paths forming a tree that connects all the nodes, having to cycle.

5.1.2.1 Multi-objective optimization model

The multi-objective optimization model can be modeled in two ways. In the first method, all of the responses are considered in the objective function. Each objective function is given a weight, and the objective function is determined for different combinations of weight factors. Formulation is given by:

$$\text{Min } \alpha\, Y_1 + \beta\, Y_2 + \gamma\, Y_3$$

where α, β and γ are the weight factors for response Y_1, Y_2 and Y_3, respectively. The weight factors are varied to obtain a different combination of objective values, and thus the behavior of each objective or performance metric can be studied.

In the second method of multi-objective optimization, only one of the three objectives is considered as the objective function, and the other two objectives are

considered as constraints. The second method is used in this research for optimization. The formulation is given as follows.

Indices

j = decision variable $(1 \ldots m)$

r = response variable $(1, 2, \text{or } 3)$

t = terminal node $(1 \ldots n_r)$

Input Data

β_{jrt} = linear regression coefficient of variable j for response r on the terminal node t

β_{0rt} = intercept of the linear regression equation for response r on the terminal node t

l_{jrt} = lower bound for variable j for response r on the terminal node t

u_{jrt} = upper bound for variable j for response r on the terminal node t

d_r = upper bound for response r

$\Omega_{r\tilde{r}}$ = the set of terminal node pairs from the trees representing response r and $\tilde{r}$ that conflict

Decision Variables

$$y_{rt} = \begin{cases} 1, \text{ if terminal node } t \text{ for the response } r \text{ is selected} \\ 0, \text{ otherwise} \end{cases}$$

x_{jrt} = value of decision variable j if terminal node t for response r is selected

x_j = value of decision variable j

Objective:

Minimize $\sum_{t=1}^{n_1} \sum_{j=1}^{m} \beta_{j1t}\, x_{j1t} + \beta_{01t} y_{1t}$

Subject to $l_{jrt}\, y_{rt} \le x_{jrt} \le u_{jrt}\, y_{rt}, \forall\, r = 1, \dots, R;$ *(1)*

$$\forall\, t = 1 \dots n_r;\ \forall j = 1 \dots m$$

$\sum_{t=1}^{n_r} y_{rt} = 1, \forall\, r = 1, \dots, R$ *(2)*

$\sum_{t=1}^{n} x_{jrt} = x_j, \forall j = 1 \dots m;\ \forall r = 1, \dots, R$ *(3)*

$\sum_{t=1}^{n_1} \sum_{j=1}^{m} \beta_{jrt}\, x_{jrt} + \beta_{0rt} y_{rt} \le d_r, \forall\, r = 2, \dots, R;\ \forall\, t = 1 \dots n_r$ *(4)*

$y_{rt} + y_{\tilde{r}\tilde{t}} \le 1;\ \ \forall (t, \tilde{t}) \in \Omega_{r\tilde{r}}, \forall r = 1, \dots, \tilde{r} - 1, \tilde{r} = 2, \dots, R$ *(5)*

Only the 32 continuous variables appear in the objective function, and hence, the categorical variables are not included in the constraints. However, some splits on the tree for responses 1 and 3 are based on categorical variables. In order to avoid conflicts and to make sure consistent terminal nodes are selected for the same categorical variable, manual inspections for categorical variable conflicts are done and are specified as constraints in the model. In this case, responses 1 and 3 have the same tree, so if terminal node 1 is selected for response 1, then terminal node 1 must also be selected for response 3. Thus, the optimization model also considers the categorical variables while selecting the best terminal node. This is given by constraint (5). The terminal nodes for the three response variables must have consistent tree logic with respect to the categorical variables. The conflicts between the terminal nodes for each response are determined by inspection before solving the optimization model.

In this formulation, the first response variable is minimized, while the other two are constrained. Specifically, the objective function corresponds to response 1, and constraint (4) limits response 2 and 3. The upper bound and the lower bound at each of the terminal nodes for each of the x variables are given by (1). The binary variable $\mathrm{y_{tr}}$

takes the value 1 if the optimization selects t as the optimum node for the response r. Constraint (2) ensures that only one terminal node is selected for each of the responses. Constraint (3) ensures that the decision variables are the same for each response tree. Constraint (5) addresses the categorical variables conflicts as explained above, between the trees that are identified in this case study.

Chapter 6

Results

6.1 Surrogate Optimization

EEPA approach is shown here to be the fastest approach when compared to pure exploration or exploitation methods. A summary of results using different methods and test functions are given in Table 6.1. Table 6.1 shows the global minimum of each of the functions, iterations taken to reach the known minimum for each method, limited to 1000 iterations. The '-' indicates that the known minimum was not reached after 1000 iterations.

As shown in Table 6-1, we see that EEPA finds the minimum in all cases, except for the Rastrigin function. Also EEPA with MARS finds the best solution faster, except for the Rosenbrock function. When EEPA with MARS failed to find the minimum solution, EEPA with RBF found the minimum quicker than other methods in the case of Rastrigin function. Also we see that exploration methods did not find the best solution, except in the case of the green building function. Exploitation managed to find the best solution for three functions but took quite long time compared to EEPA. For the green building function since the sampling method used is a mixed array because of discrete numeric and binary variables, the Sobol' sequence is not applicable.

Figures 6-1 to 6-5 shows how close each method came to the best solution.

Table 6-1 Results Summary

Function	Dimension	Global Minimum	Minimum after 1000 iterations						
			Maximin	Cosine	Monte Carlo	Sobol'	Exploitation	EEPA	
								MARS	RBF
Levy	20	0	-	-	-	-	-	**414**	530
Rastrigin	30	0	-	-	-	-	731	-	**92**
Rosenbrock	16	0	-	-	-	-	139	283	**31**
Sphere	27	0	-	-	-	-	-	**340**	-
Green Building	30	164	95	53	226	NA	731	**45**	**45**

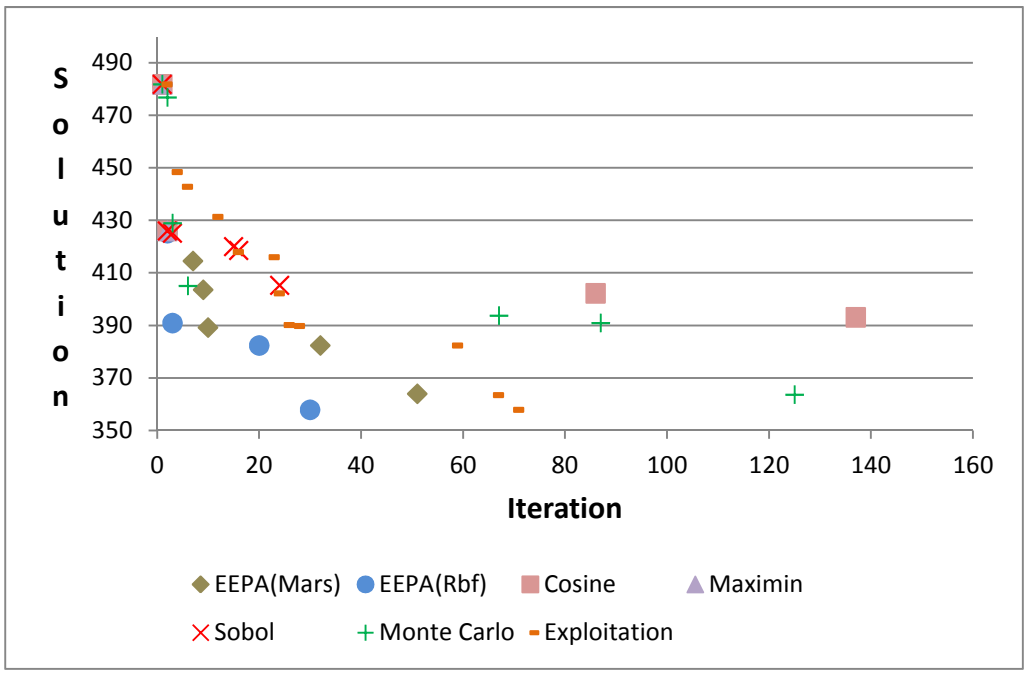

Figure 6-1 Rastrigin function – Results

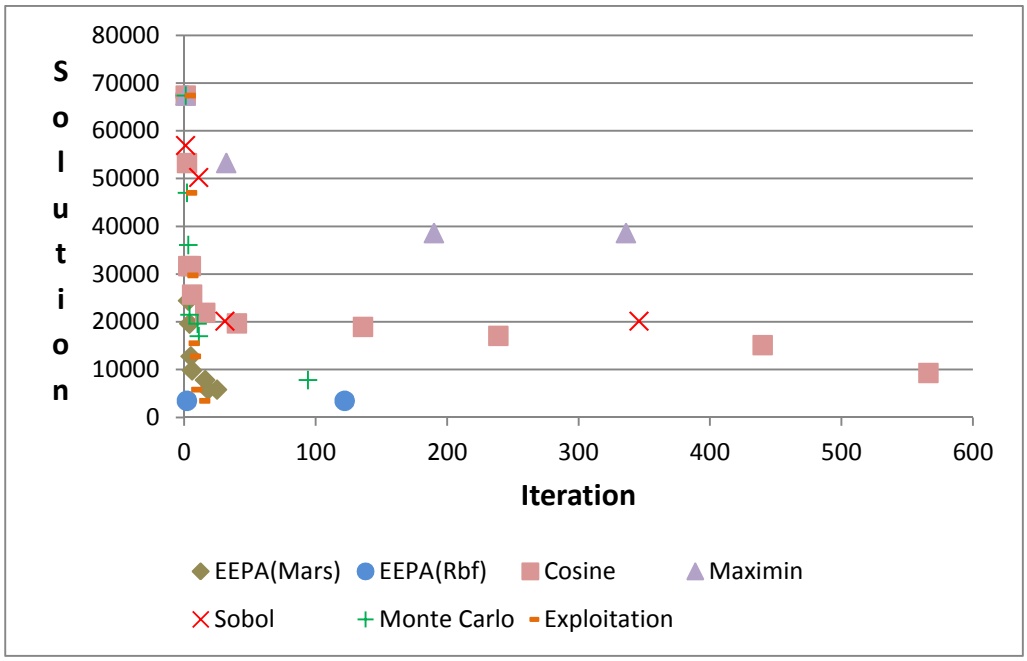

Figure 6-2 Rosenbrock function- Results

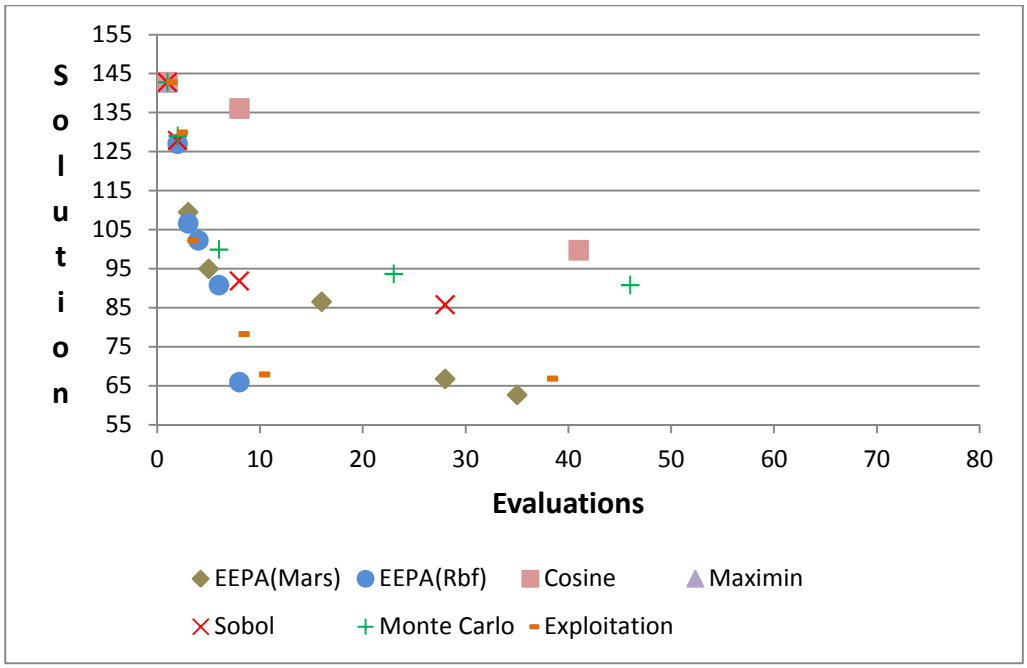

Figure 6-3 Levy function – Results

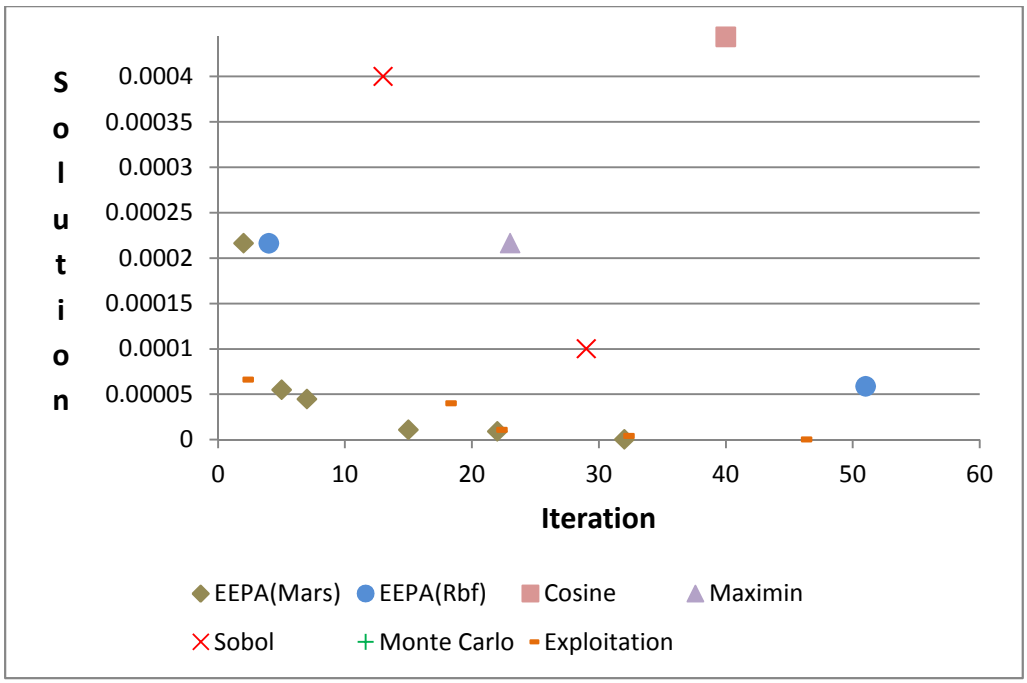

Figure 6-4 Sphere function – Results

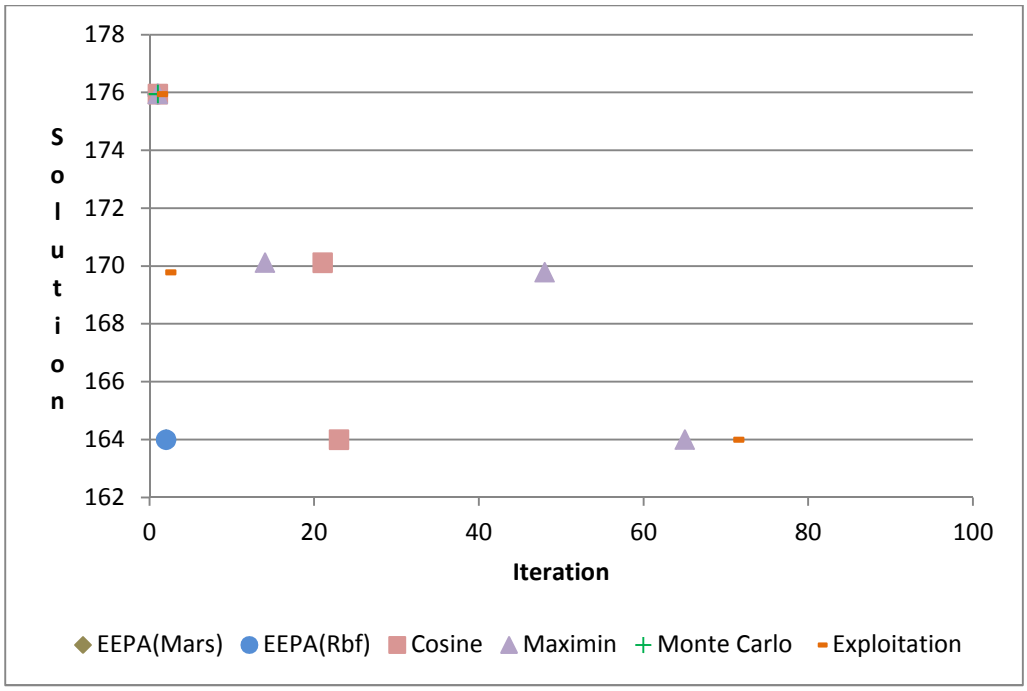

Figure 6-5 Green Building function - Results

It is also noted from these graphs that among the exploration methods, the cosine method performed well. The cosine method came closer to the best solution than other exploration methods like maximin, Monte Carlo and Sobol′. For example in the case of the Levy function (Figure 6-3) the cosine method came very close to the known minimum, although it took some time, whereas the maximin or Monte Carlo methods never found the known minimum. Figure 6-5 shows the results for the green building function. Since treed regression was used as the true function, the function had flat regions, and so the solutions were reached faster. Almost all the methods found the best solution, except for the Monte Carlo method. Hence, the speed depends on the function and its complexity.

It should be noted that the function also depends on the specified range of the input space. To test the sensitivity to the input range, EEPA was also tested for several different ranges for the Levy function, since the function is defined on $(-\infty, \infty)$. Twenty different ranges were obtained using the Sobol′ sequence as given in Table 6-2.

Table 6-2 Ranges for Levy Function

Case	Lower Bound	Upper Bound
1	-5	5.5
2	-2.5	3.25
3	-7.5	7.75
4	-6.25	4.375
5	-1.25	8.875
6	-3.75	2.125
7	-8.75	6.625
8	-8.125	3.8125
9	-3.125	8.3125
10	-0.625	1.5625
11	-5.625	6.0625
12	-6.875	2.6875
13	-1.875	7.1875
14	-4.375	4.9375
15	-9.375	9.4375
16	-9.0625	5.21875
17	-4.0625	9.71875
18	-1.5625	2.96875
19	-6.5625	7.46875
20	-5.3125	1.84375
21	-0.3125	6.34375

The results are shown in Table 6-3. Comparisons were made between EEPA, pure exploration and pure exploitation methods. As seen from the table, EEPA found the

best solution faster on 15 occasions out of 21, of which EEPA with MARS found the solution 7 times, and EEPA with RBF found it 8 times. Overall, EEPA dominated the other methods. Also from Figures 6-6 to 6-26 we can see that both EEPA implementations reached the best solution in almost all the cases. The cosine method did well among the exploration methods by finding the solution fastest on three occasions. Also, from Figures 6-6 to 6-26, we can see that the cosine method came closer to the best solution in most of the cases, whereas other pure exploration methods did not. The pure exploitation method was also the fastest to find the solution on three occasions . In most of the cases, the total number of function evaluations for EEPA to reach the best solution was fewer than 300. Note that the horizontal axis in Figures 6-6 to 6-26 correspond to each iteration of respective methods, in which 10 points and functions evaluations were added.

Table 6-3 Results for multiple ranges – Levy function

Method	**Number of times fastest solution found**
EEPA (Mars)	7
EEPA (Rbf)	8
Cosine	3
Maximin	-
Sobol’	-
Monte Carlo	-
Pure Exploitation	3

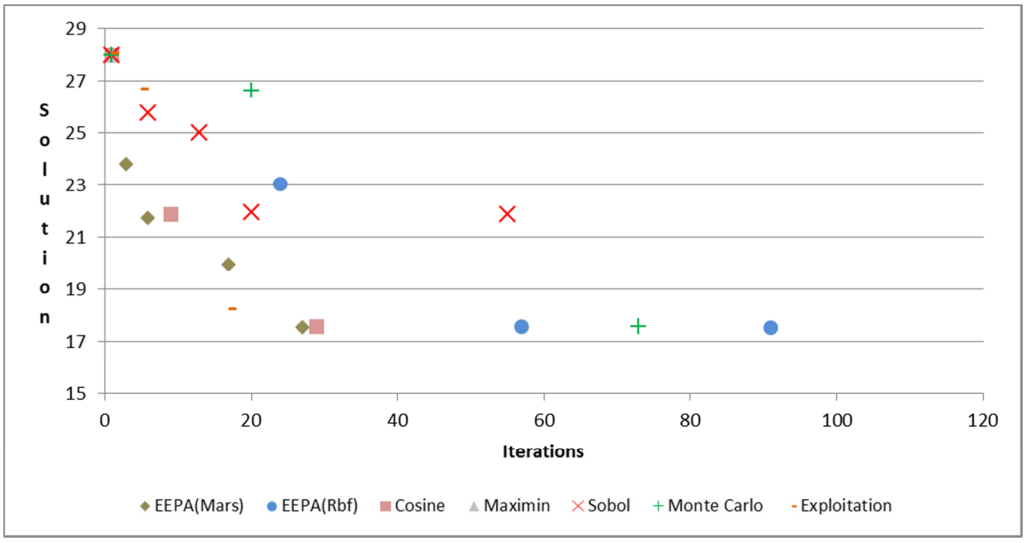

Figure 6-6 Range (-5, 5.5)

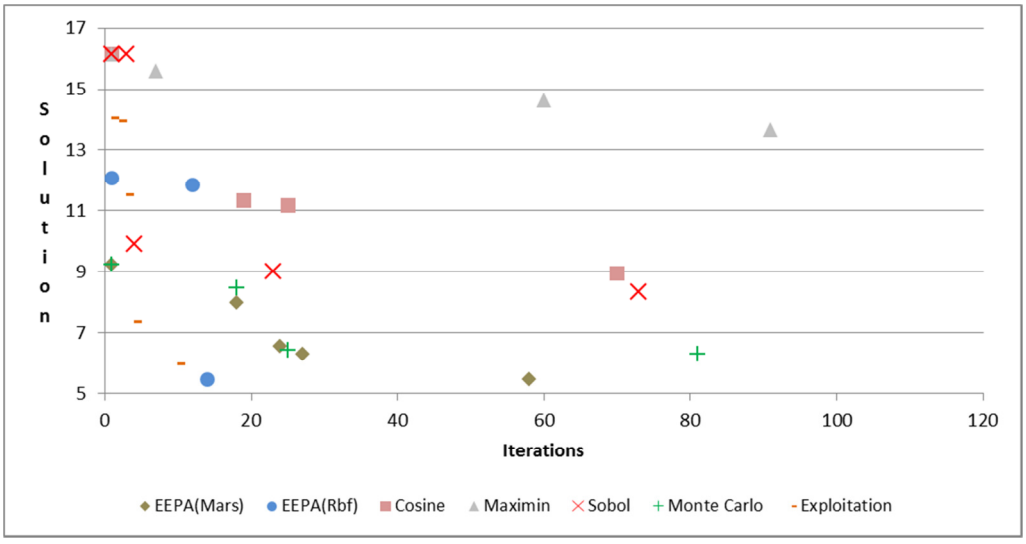

Figure 6-7 Range (-2.5, 3.25)

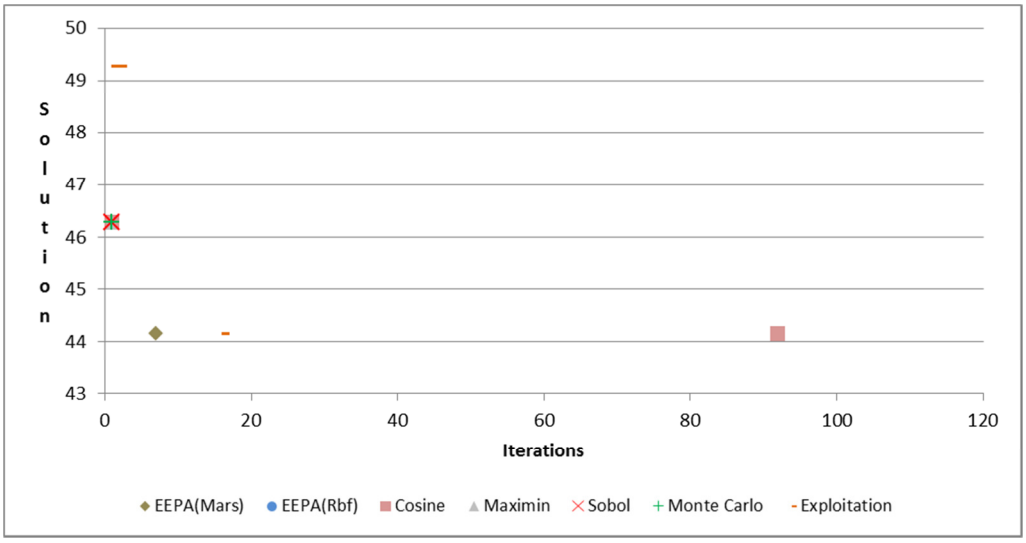

Figure 6-8 Range (-7.5, 7.75)

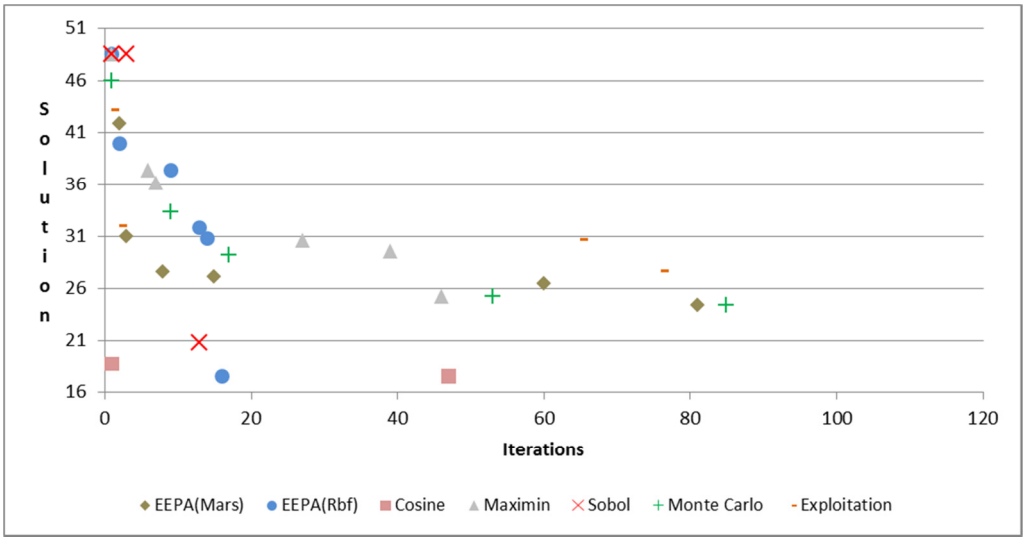

Figure 6-9 Range (-6.25, 4.375)

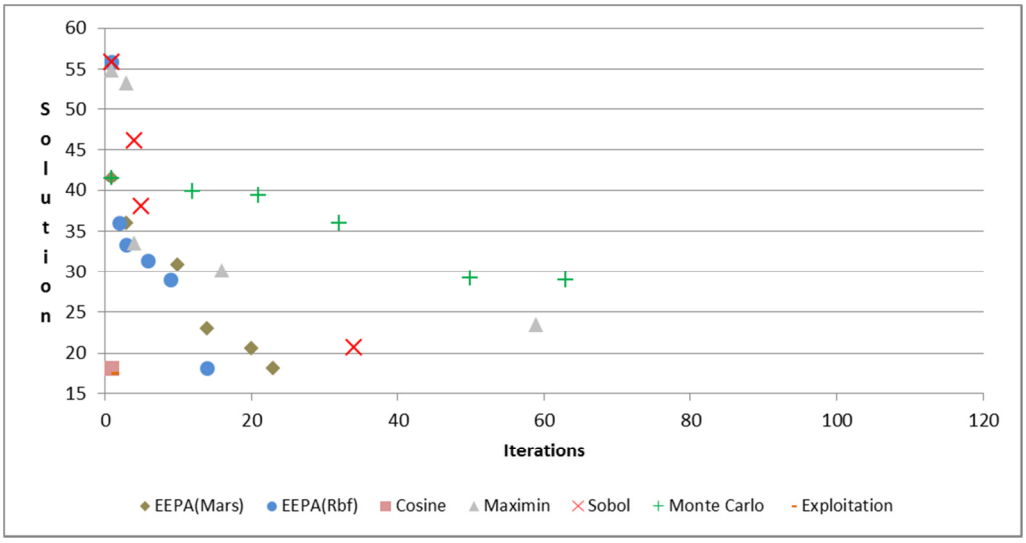

Figure 6-10 Range (-1.25, 8.875)

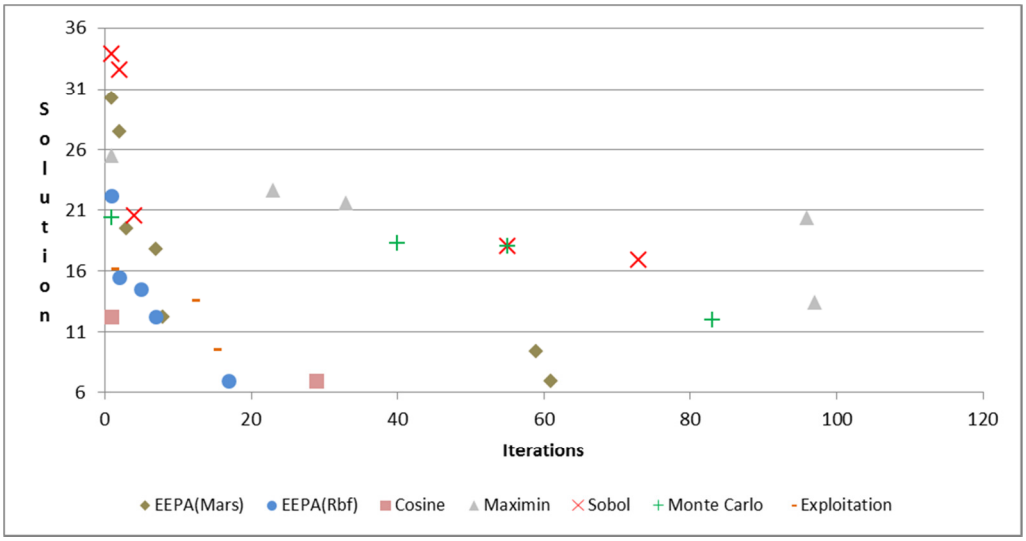

Figure 6-11 Range (-3.75, 2.125)

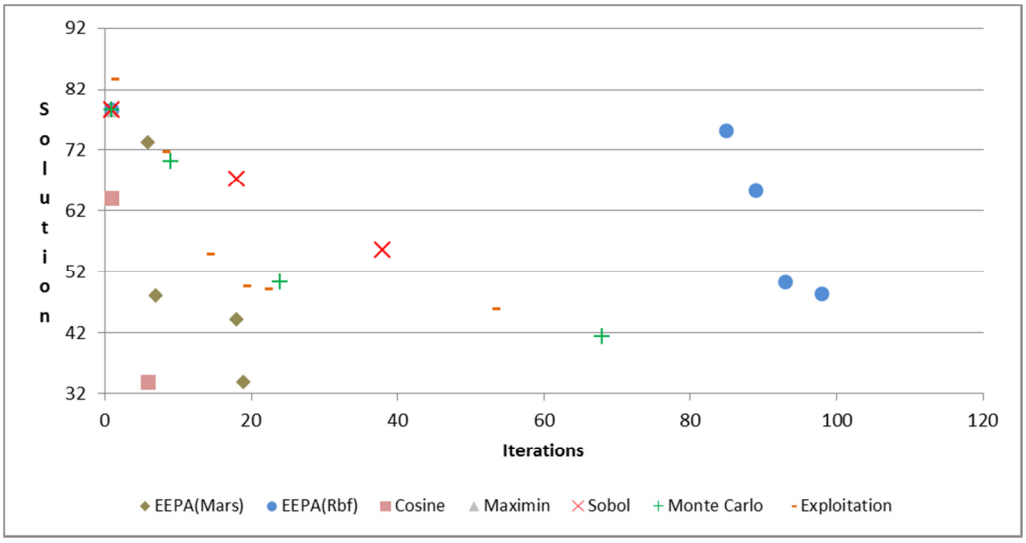

Figure 6-12 Range (-8.75, 6.625)

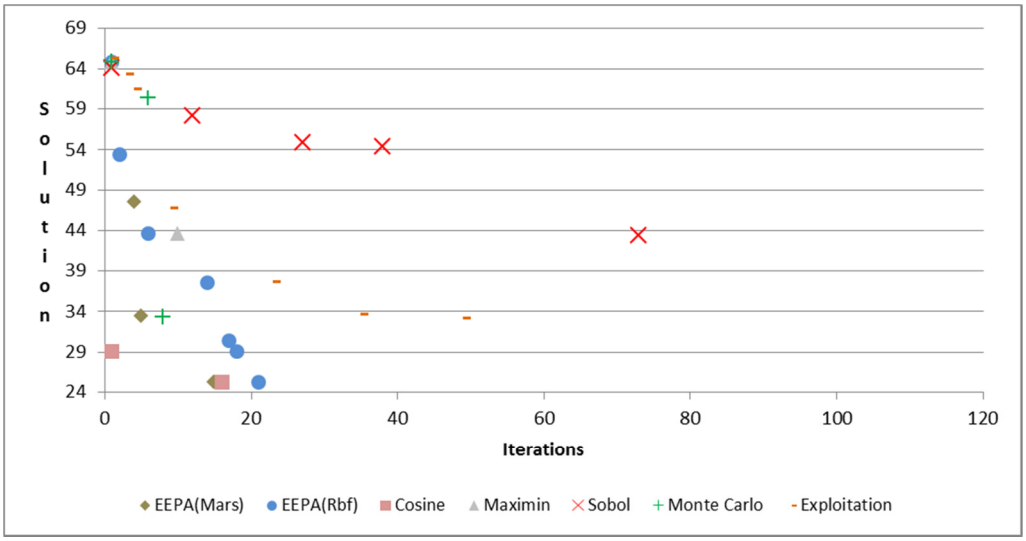

Figure 6-13 Range (-8.125, 3.8125)

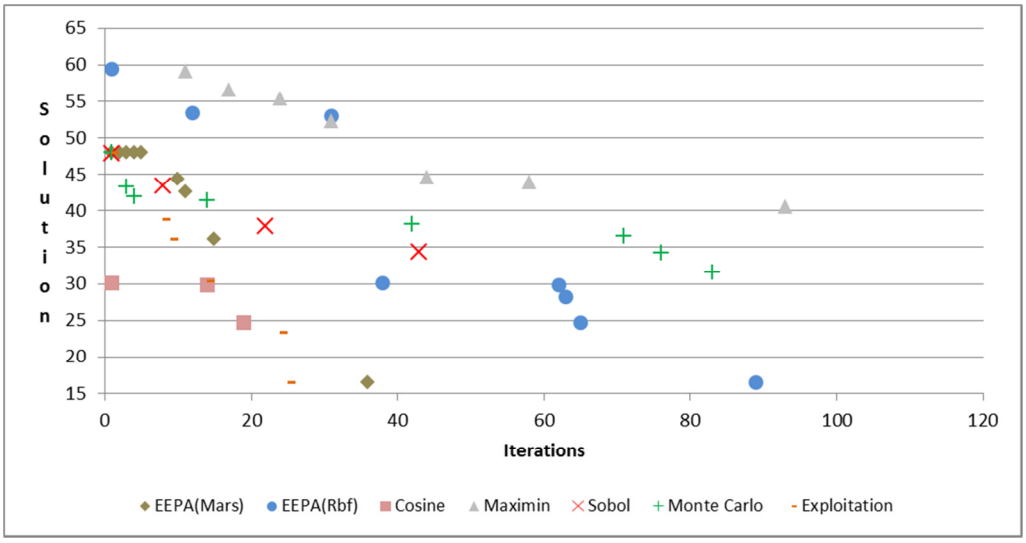

Figure 6-14 Range (-3.125, 8.3125)

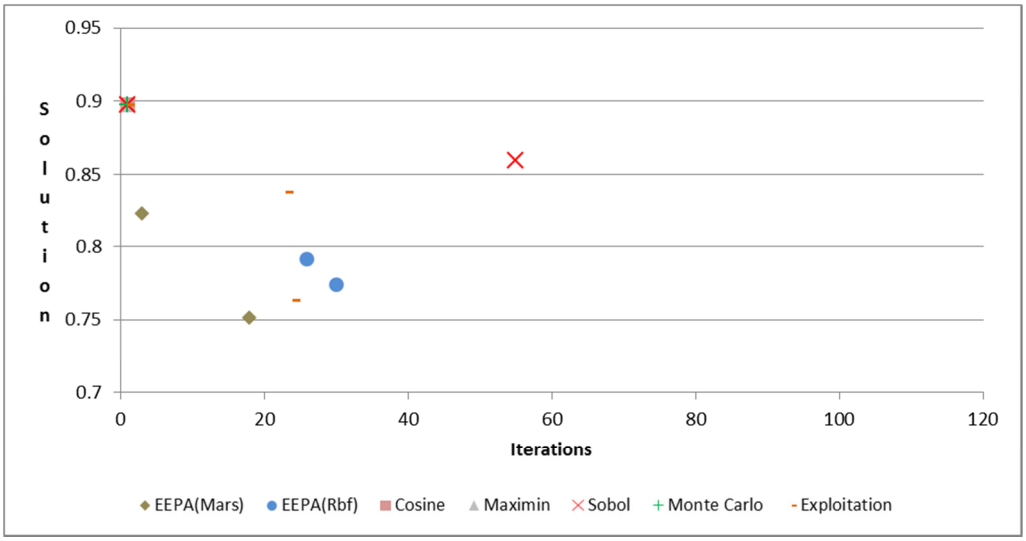

Figure 6-15 Range (-0.625, 1.5625)

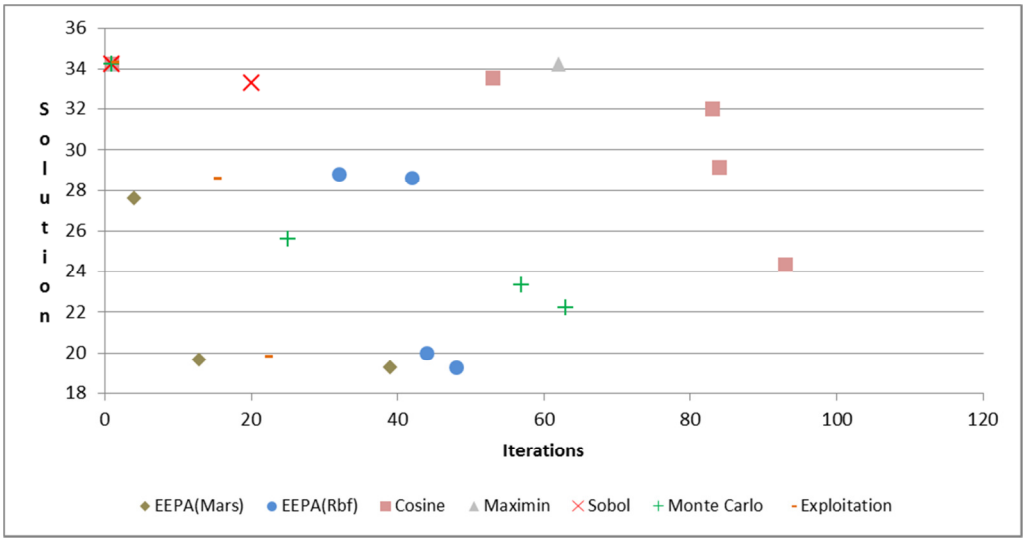

Figure 6-16 Range (-5.625, 6.0625)

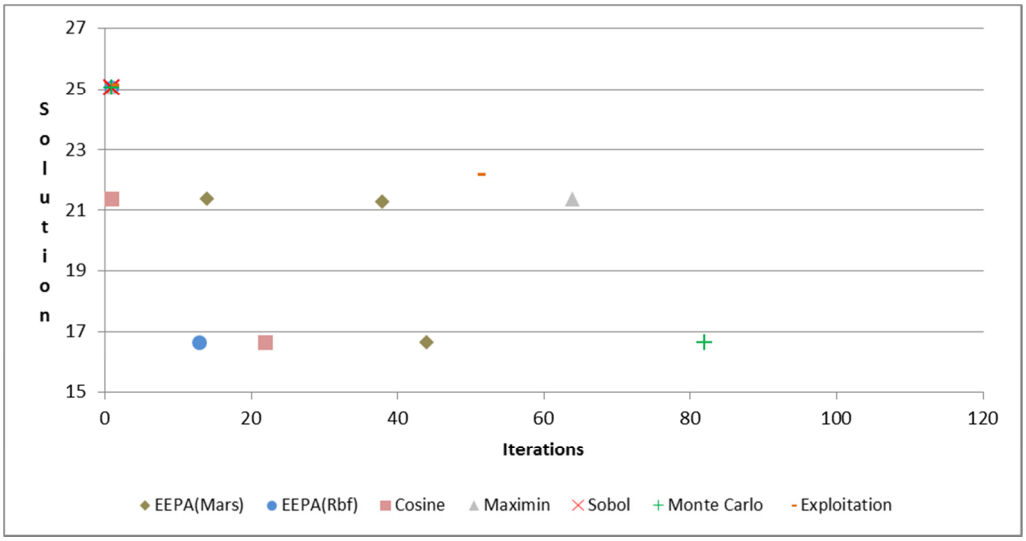

Figure 6-17 Range (-6.875, 2.6875)

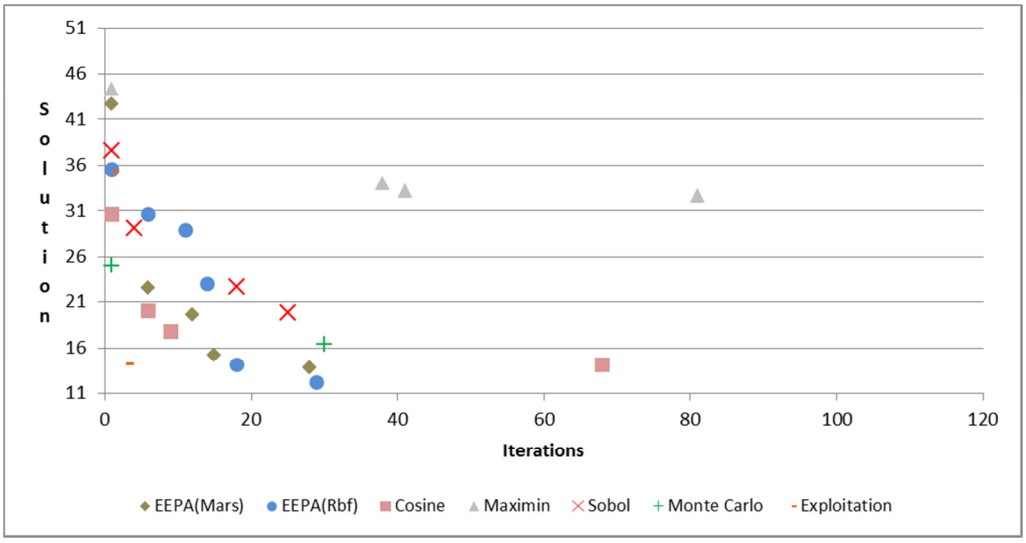

Figure 6-18 Range (-1.875, 7.1875)

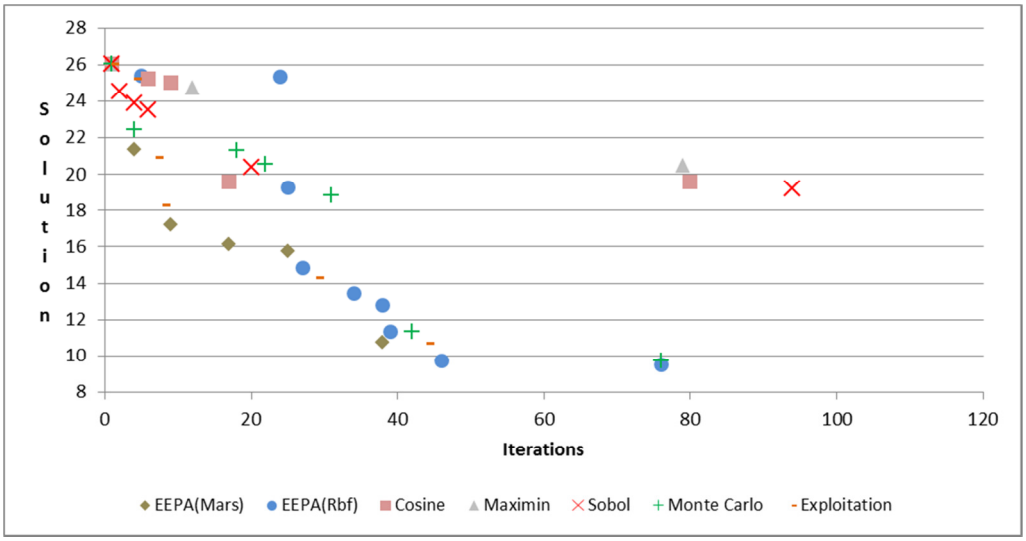

Figure 6-19 Range (-4.375, 4.9375)

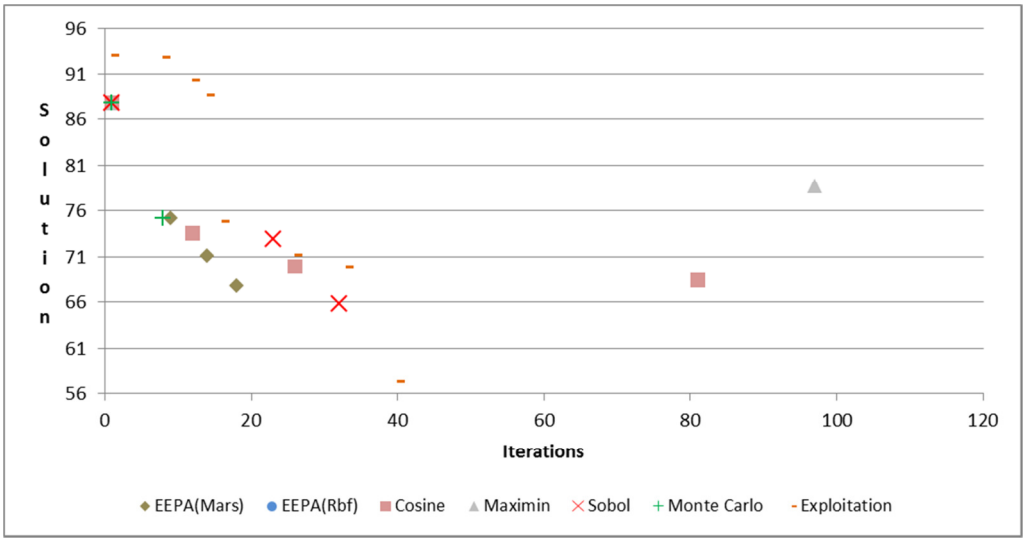

Figure 6-20 Range (-9.375, 9.4375)

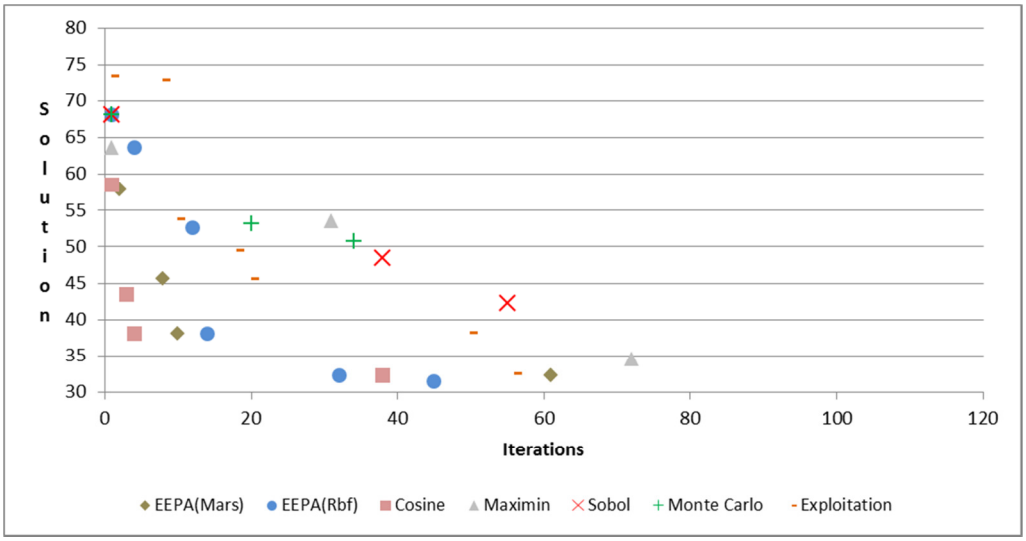

Figure 6-21 Range (-9.0625, 5.21875)

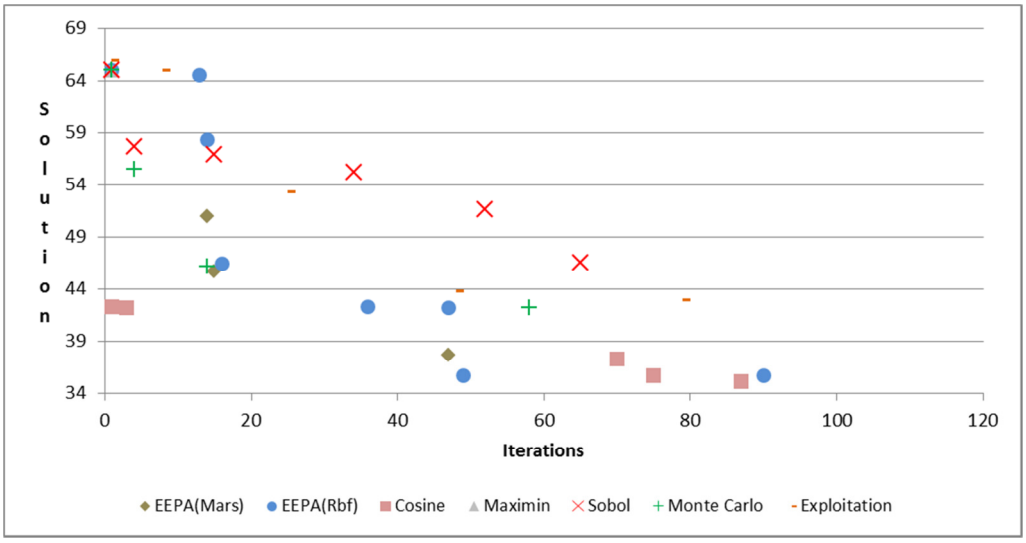

Figure 6-22 Range (-4.0625, 9.71875)

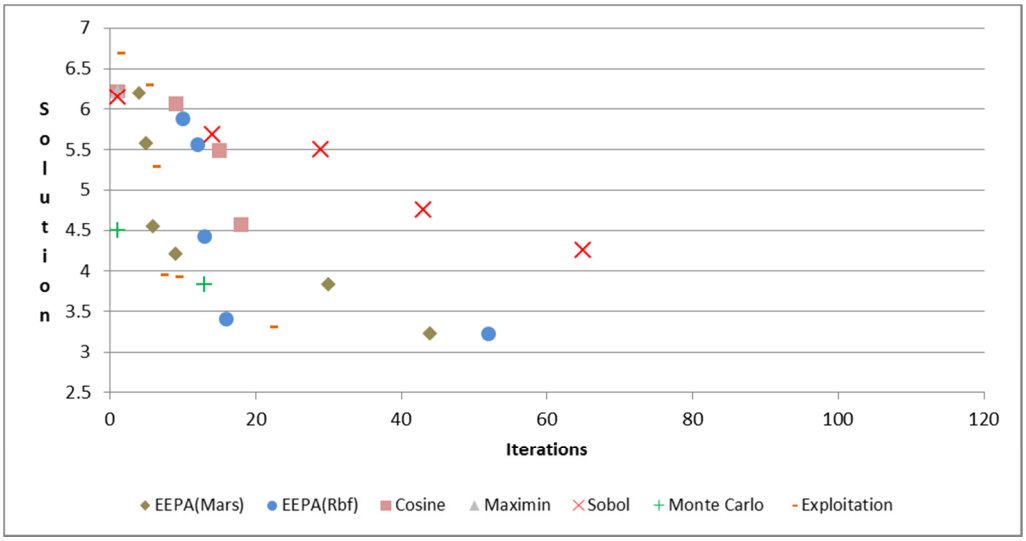

Figure 6-23 Range (-1.5625, 2.96875)

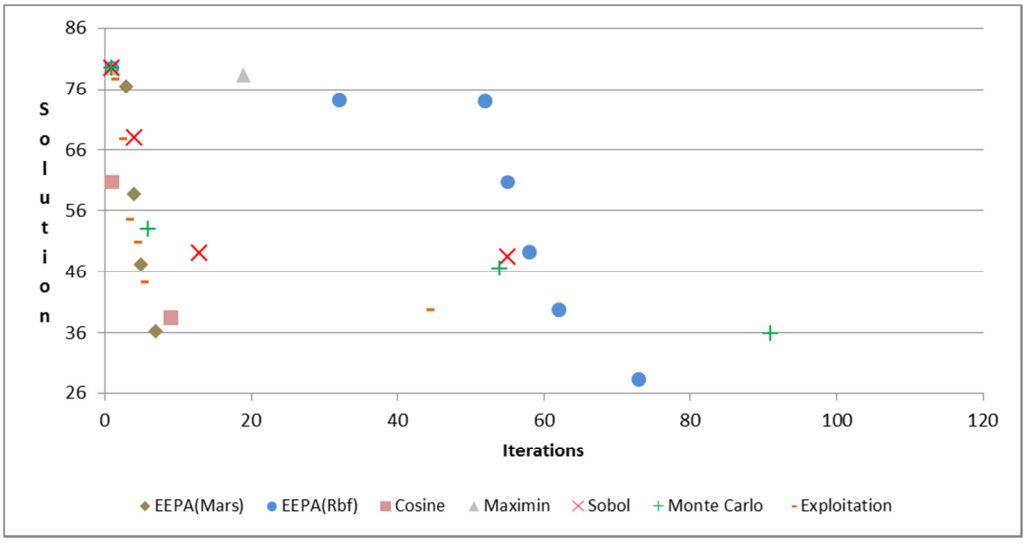

Figure 6-24 Range (-6.5625, 7.46875)

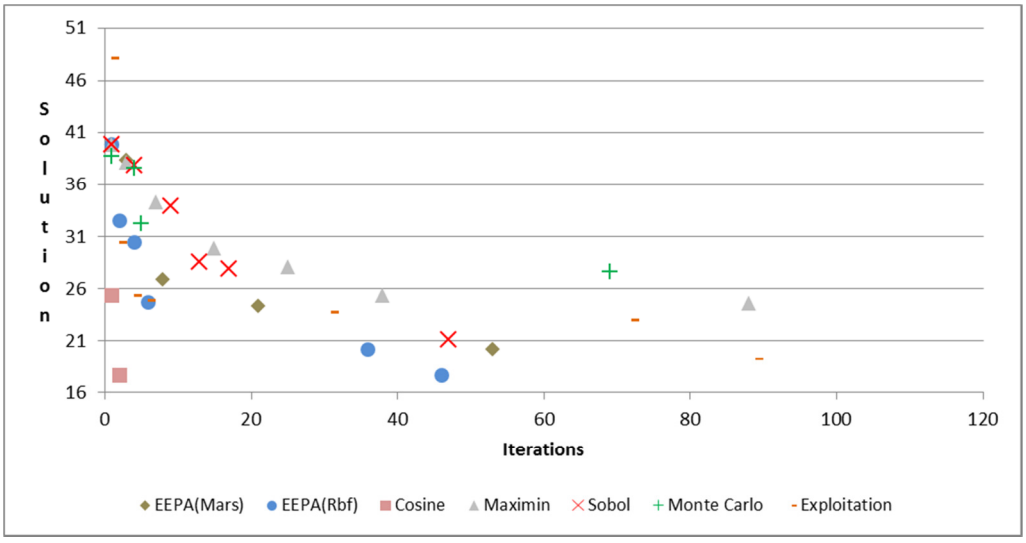

Figure 6-25 Range (-5.3125, 1.84375)

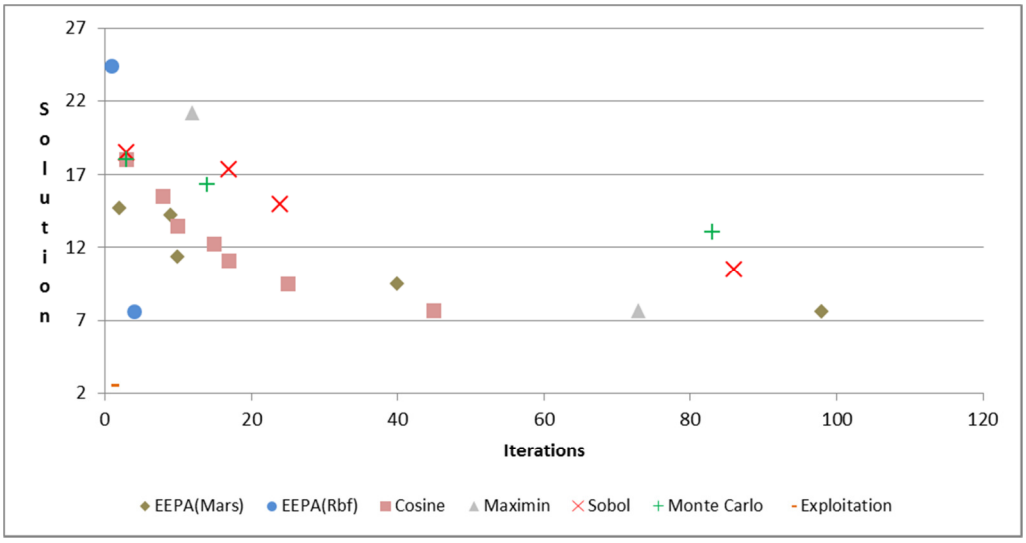

Figure 6-26 Range (-0.3125, 6.34375)

6.2 Optimization of Tree Based Model

The optimization was conducted using Xpress optimization suite. All three objectives were optimized separately using the multi-response metamodel shown in Figure 5-1 to 5-3. The objective is to minimize all three responses, namely annual source total energy, HVAC energy usage and total utility cost.

A unique optimal solution is found in terminal node 1 when the objective is to minimize annual source total energy. At this terminal node only 26 of the 32 continuous variables are significant with non-zero coefficients in the regression model. The values for the insignificant variables can be arbitrarily selected within their defined ranges. The minimum number of windows and minimum occupant density were selected while minimizing the annual source energy. In addition, the maximum amount of exterior insulation and the minimum amount of additional wall insulation were selected, which implies a balance is established between the insulations.

For HVAC energy (response 2), result showed that terminal node 7 is optimum in which 12 of the 32 continuous decision variables have non-zero coefficients in the regression model. The minimum HVAC energy is obtained by utilizing the minimum number of windows and the minimum amount of occupant density. In addition, the maximum amount of exterior floor insulation and exterior roof insulation are used.

The results also showed that terminal node 3 includes a unique optimal solution for annual total utility cost. At this terminal node, 17 of the 32 decision variables were significant in the regression model. The maximum amount of floor insulation, windows and vents are selected by the optimization to minimize the annual total utility cost.

The results are summarized in Table 6-3, which shows that for some variables the highest value is selected and for some variables the lowest value is selected within their ranges for each of the three responses. The insignificant variables are denoted using '*' in the table. The significant categorical variables are obtained by traversing through the optimum terminal node, is also shown in Table 6-3.

Table 6-4 Building decision variable and optimization results

46 Input Variables	**Type**	**Y1**	**Y2**	**Y3**
Construction (x1)	Numerical	Lowest	*	Highest
Interior Insulation (x2)	Numerical	Highest	Highest	*
Cap (x3)	Numerical	Lowest	*	Lowest
Exterior Cavity Insulation (x4)	Numerical	Highest	Highest	Highest
Exterior Wall Insulation (x5)	Numerical	Lowest	*	Highest
Additional Wall Insulation (x6)	Numerical	Highest	*	*
%Window-North (x7)	Numerical	Lowest	Lowest	Highest
%Window-South (x8)	Numerical	*	*	*
%Window-East (x9)	Numerical	Lowest	*	*
%Window-West (x10)	Numerical	*	*	*
Additional Roof Insulation (x11)	Numerical	Lowest	*	*
Ceiling Batt Insulation (x12)	Numerical	Highest	*	*
Exterior Roof Insulation (x13)	Numerical	Highest	*	*
Footprint X&Y (x14)	Numerical	Lowest	Highest	Highest
Dimension-Width (x15)	Numerical	Lowest	*	Lowest
Frame Width (x16)	Numerical	*	Lowest	*
Design Max Occupant Density-	Numerical	Lowest	Highest	Highest

Table 6-4—*Continued*

Residential (General Living Space) (x17)				
Design Ventilation-Residential (General Living Space) (x18)	Numerical	*	*	*
Design Max Occupant Density-Residential (Bedroom) (x19)	Numerical	Lowest	*	Highest
Design Ventilation-Residential (Bedroom) (x20)	Numerical	Highest	*	*
Design Max Occupant Density-Residential (Garage) (x21)	Numerical	Lowest	*	Highest
Design Ventilation-Residential (Garage) (x22)	Numerical	*	*	*
Design Max Occupant Density- Dining Area (x23)	Numerical	*	*	*
Design Ventilation-Dining Area (x24)	Numerical	Highest	*	Highest
Design Max Occupant Density-Kitchen and Food Preparation (x25)	Numerical	*	Lowest	*
Design Ventilation-Kitchen and Food Preparation (x26)	Numerical	Highest	*	Lowest
Design Max Occupant Density-Corridor (x27)	Numerical	*	*	*
Design Ventilation-Corridor (x28)	Numerical	Highest	Lowest	Highest
Design Max Occupant Density-Laundry (x29)	Numerical	Lowest	*	Lowest
Design Ventilation-Laundry (x30)	Numerical	Lowest	Lowest	Highest
Design Max Occupant Density-All Others (x31)	Numerical	Lowest	Lowest	Highest
Design Ventilation-All Others (x32)	Numerical	Lowest	*	Highest
Wall Construction (x33)	Categorical	*	*	*
Windows-Glass Category (x34)	Categorical	*	*	*
Roof Construction (x35)	Categorical	*	*	*
Exterior Wall Finishes (x36)	Categorical	*	*	Brick
Exterior Wall Color (x37)	Categorical	*	*	*
Interior Wall Insulation (x38)	Categorical	None	*	1-inch
Exterior Roof Finish (x39)	Categorical	*	*	*
Exterior Roof Color (x40)	Categorical	*	*	*
Doors-Construction (x41)	Categorical	*	*	*
Pitched Roof (x42)	Categorical	*	*	*
Ceiling Interior Finishes (x43)	Categorical	*	*	*
Windows-Glass Type (x44)	Categorical	*	*	*
Orientation (x45)	Categorical	*	*	*
Doors-Glass Type (x46)	Categorical	*	*	*
Optimal Objective Value		**658 Mbtu**	**12 Mbtu**	**$11340**

.

Chapter 7

Conclusions and Future Research

The surrogate-based optimization using exploration and exploitation looks promising. EEPA is shown to be the overall best method to minimize the number of expensive, time consuming computer model runs. The EEPA method performs better than pure exploration and exploitation methods. Surrogate optimization is metamodel independent when EEPA is used. Though two different criteria are used to select the points, a Pareto frontier instead of weights to simplify and automate the process. In EEPA, maximin approach is used as the exploration metric and predicted response as the exploitation metric, but certainly other exploration metric like cosine and any other exploitation metric could also be used.

The EEPA process, presented in Chapter 3 is shown to be efficient. It is also understood that the minimum depends on the existence of that point in the randomly generated set R, which could be a drawback for the current implementation of EEPA because the point's existence is not guaranteed when points are generated randomly.

Future research would be to develop a formal optimization routine to select points from the input space instead of defining a random data set. An optimal way of selecting points from the Pareto frontier can also be studied instead of using an exploration process. This study can be compared with other surrogate optimization studies performed using weight factors [53]. The study can also be extended to address the complete green building application, which was briefly described in Chapter 5, by incorporating all the categorical variables and multiple objectives. This study can also be extended to other applications where computer experiments are used.

References

[1] Alexander, W. P., & Grimshaw, S. D. (1996). Treed regression. *Journal of Computational and Graphical Statistics*, *5*(2), 156-175.

[2] ATHENA Impact Estimator for buildings, V. 4.1.13 (2011). The ATHENA Institute, Canada

[3] Atkinson, A. C., & Fedorov, V. V. (1975). Optimal design: experiments for discriminating between several models. Biometrika, 62(2), 289-303.

[4] Azadivar, F. (1999, December). Simulation optimization methodologies. In Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future-Volume 1 (pp. 93-100). ACM.

[5] Azadivar, F., & Tompkins, G. (1999). Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. European Journal of Operational Research, 113(1), 169-182.

[6] BEES Online (2010). National Institute of Standards and Technology, USA

[7] Bian, W., & Tao, D. (2011). Max-min distance analysis by using sequential SDP relaxation for dimension reduction. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 33(5), 1037-1050.

[8] Booker, A. J., Dennis Jr, J. E., Frank, P. D., Serafini, D. B., Torczon, V., & Trosset, M. W. (1999). A rigorous framework for optimization of expensive functions by surrogates. Structural optimization, 17(1), 1-13.

[9] Box, G. E., & Draper, N. R. (1987). Empirical model-building and response surfaces. John Wiley & Sons.

[10] Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and regression trees. Wadsworth & Brooks. Monterey, CA.

[11] Building Energy Software Tools Directory (2011). Energy Efficiency and Renewable Energy (EERE), USA. http://www.eere.energy.gov/ (accessed December, 2011)

[12] Carson, Y., & Maria, A. (1997, December). Simulation optimization: methods and applications. In Proceedings of the 29th conference on Winter simulation (pp. 118-126). IEEE Computer Society.

[13] Censor, Y. (1977). Pareto optimality in multiobjective problems. Applied Mathematics and Optimization, 4(1), 41-59.

[14] Cheng, B., & Titterington, D. M. (1994). Neural networks: A review from a statistical perspective. Statistical science, 2-30.

[15] Chernoff, H. (1959). Sequential design of experiments. The Annals of Mathematical Statistics, 30(3), 755-770.

[16] Corley, H. W., Rosenberger, J., Yeh, W. C., & Sung, T. K. (2006). The cosine simplex algorithm. *The International Journal of Advanced Manufacturing Technology*, *27*(9-10), 1047-1050.

[17] Crombecq, K., Gorissen, D., Deschrijver, D., & Dhaene, T. (2011). A novel hybrid sequential design strategy for global surrogate modeling of computer experiments. SIAM Journal on Scientific Computing, 33(4), 1948-1974.

[18] Currin, C., Mitchell, T., Morris, M., & Ylvisaker, D. (1991). Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. Journal of the American Statistical Association, 86(416), 953-963.

[19] Davis, L. (Ed.). (1991). Handbook of genetic algorithms (Vol. 115). New York: Van Nostrand Reinhold.

[20] D. R. Jones, “A taxonomy of global optimization methods based on response surfaces,” Journal of Global Optimization, vol. 21, pp. 345–383, 2001

[21] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” Journal of Global Optimization, vol. 13, pp. 455–492, 1998

[22] Dyn, N., Levin, D., & Rippa, S. (1986). Numerical procedures for surface fitting of scattered data by radial functions. SIAM Journal on Scientific and Statistical Computing, 7(2), 639-659.

[23] eQUEST, V. 3.63b (2009). James J. Hirsch and Associates, USA.

[24] Farhang-Mehr, A., & Azarm, S. (2005). Bayesian meta-modelling of engineering design simulations: a sequential approach with adaptation to irregularities in the response behaviour. International Journal for Numerical Methods in Engineering, 62(15), 2104-2126.

[25] Forrester, A. I., & Keane, A. J. (2009). Recent advances in surrogate-based optimization. Progress in Aerospace Sciences, 45(1), 50-79.

[26] Friedman, J. H. (1991). Multivariate adaptive regression splines. *The annals of statistics*, 1-67..

[27] G. Wang and S. Shan, “Review of metamodeling techniques in support of engineering design optimization,” ASME Transactions, Journal of Mechanical Design, vol. 129, no. 4, pp. 370–380, 2007

[28] Gehrke, J., Ganti, V., Ramakrishnan, R., & Loh, W. Y. (1999, May). BOAT-optimistic decision tree construction. In SIGMOD Conference (Vol. 12).

[29] Giunta, A. A., Watson, L. T., & Koehler, J. (1998, September). A comparison of approximation modeling techniques: polynomial versus interpoLating models. In Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Design (pp. 392-404).

[30] Gramacy, R. B., & Lee, H. K. (2009). Adaptive design and analysis of supercomputer experiments. Technometrics, 51(2), 130-145

[31] Guéret, C., Prins, C., & Sevaux, M. (1999). Applications of optimization with Xpress-MP. contract, 00034.

[32] H.-M. Gutmann, “A radial basis function method for global optimization,” Journal of Global Optimization, vol. 19, no. 3, pp. 201–227, 2001

[33] Halton, J. H. (1964). Algorithm 247: Radical-inverse quasi-random point sequence. Communications of the ACM, 7(12), 701-702.

[34] Hardy, R. L. (1971). Multiquadric equations of topography and other irregular surfaces. Journal of geophysical research, 76(8), 1905-1915.

[35] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., & Tibshirani, R. (2009). *The elements of statistical learning* (Vol. 2, No. 1). New York: Springer

[36] Hedayat, A. S., Sloane, N. J. A., & Stufken, J. (1999). Orthogonal arrays: theory and applications. Springer.

[37] Henri Faure,(1992), "Good permutations for extreme discrepancy", Journal of Number

[38] Huang, D., Allen, T. T., Notz, W. I., & Miller, R. A. (2006). Sequential kriging optimization using multiple-fidelity evaluations. Structural and Multidisciplinary Optimization, 32(5), 369-382.

[39] Jeffers, J. N. R. (1978). Design of experiments, NERC.

[40] Jin, R., Chen, W., & Simpson, T. W. (2001). Comparative studies of metamodelling techniques under multiple modelling criteria. Structural and Multidisciplinary Optimization, 23(1), 1-13.

[41] Jin, R., Chen, W., & Sudjianto, A. (2002, September). On sequential sampling for global metamodeling in engineering design. In Proceedings of DETC (Vol. 2, pp. 539-548).

[42] Johnson, M. E., Moore, L. M., & Ylvisaker, D. (1990). Minimax and maximin distance designs. Journal of statistical planning and inference, 26(2), 131-148.

[43] Kleijnen, J. P. (1988). Analyzing simulation experiments with common random numbers. Management science, 34(1), 65-74.

[44] Kleijnen, J. P., & Van Beers, W. C. (2004). Application-driven sequential designs for simulation experiments: Kriging metamodelling. Journal of the Operational Research Society, 55(8), 876-883.

[45] Koehler, J. R., & Owen, A. B. (1996). Computer experiments. Handbook of statistics, 13(13), 261-308.

[46] Kung, P. (2012). “Multivariate Modeling For A Multiple Stage, Multiple Objective Green Building Framework” Ph.D. Dissertation, The University of Texas at Arlington

[47] Lei, G., Shao, K. R., Guo, Y., Zhu, J., & Lavers, J. D. (2008). Sequential optimization method for the design of electromagnetic device. Magnetics, IEEE Transactions on, 44(11), 3217-3220.

[48] M. Bjorkman and K. Holmstrom, “Global optimization of costly non-convex functions using radial basis functions,” Journal of Global Optimization, vol. 19, pp. 201–227, 2001.

[49] M. Emmerich, A. Giotis, M. Ozdemir, T. Back, and K. Giannakoglou, “Metamodel assisted evolution strategies,” Parallel Problem Solving from Nature V (PPSN VII), Springer-Verlag, pp. 362–370, 2002

[50] Martinez, N.M. (2013). “Global optimization of nonconvex piecewise linear regression splines” Ph.D. Dissertation, The University of Texas at Arlington.

[51] McKay, M. D., Bechman, R. J., and Conover, W. J., 1979, “A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code,” Technometrics, 21(2), May, pp. 239–245.

[52] Müller, J., & Piché, R. (2011). Mixture surrogate models based on Dempster-Shafer theory for global optimization problems. Journal of Global Optimization, 51(1), 79-104.

[53] Müller, J., & Shoemaker, C. A. (2014). Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems. Journal of Global Optimization, 1-22.

[54] Myers, R. H., & Montgomery, D. C. Response Surface Methodology,(1995). Willey Inter-Science.

[55] Neelakantan, T. R., & Pundarikanthan, N. V. (2000). Neural network-based simulation-optimization model for reservoir operation. Journal of water resources planning and management, 126(2), 57-64.

[56] Niederreiter, H. (1992). Quasi-Monte Carlo Methods. John Wiley & Sons, Ltd.

[57] Osio, I. G., & Amon, C. H. (1996). An engineering design methodology with multistage Bayesian surrogates and optimal sampling. Research in Engineering Design, 8(4), 189-206.

[58] Provost, F., Jensen, D., & Oates, T. (1999, August). Efficient progressive sampling. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 23-32). ACM.

[59] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., & Kevin Tucker, P. (2005). Surrogate-based analysis and optimization. Progress in Aerospace Sciences, 41(1), 1-28.

[60] R. Jin, W. Chen, and T. Simpson, “Comparative studies of meta-modeling techniques under multiple modeling criteria,” Journal of Structural and Multidisciplinary Optimization, vol. 23, no. 1, pp. 1–13, 2001

[61] Regis, R. G., & Shoemaker, C. A. (2005). Constrained global optimization of expensive black box functions using radial basis functions. Journal of Global optimization, 31(1), 153-171.

[62] Regis, R. G., & Shoemaker, C. A. (2007a). A stochastic radial basis function method for the global optimization of expensive functions. INFORMS Journal on Computing, 19(4), 497-509

[63] Regis, R. G., & Shoemaker, C. A. (2007b). Parallel radial basis function methods for the global optimization of expensive functions. European journal of operational research, 182(2), 514-535.

[64] Regis, R. G., & Shoemaker, C. A. (2013). Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. Engineering Optimization, 45(5), 529-555.

[65] S. Crino and D. E. Brown, “Global optimization with multivariate adaptive regression splines,” IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, vol. 37, no. 2, 2007

[66] S. Hosder, L. T. Watson, B. Grossman, W. H. Mason, H. Kim, R. T. Haftka, and S. E. Cox, Polynomial Response Surface Approximations for the Multidisciplinary Design Optimization of a High Speed Civil Transport. Dordrecht, The Netherlands: Kluwer Academic Publishers, 2001, vol. 2

[67] Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. Statistical science, 4(4), 409-423.

[68] Sahu, S. (2011). “Multivariate adaptive regression spline based framework for statistically parsimonious adaptive dynamic programming,” Ph.D. Dissertation, The University of Texas at Arlington.

[69] Shannon, C. E., & Weaver, W. (1948). A mathematical theory of communication.

[70] Simpson, T. W., Allen, J. K., & Mistree, F. (1998a, September). Spatial correlation metamodels for global approximation in structural design optimization. In ASME Design Engineering Technical Conferences.

[71] Simpson, T. W., Mauery, T. M., Korte, J. J., & Mistree, F. (1998b). Comparison of response surface and kriging models for multidisciplinary design optimization. AIAA paper 98, 4758(7).

[72] Simpson, T. W., Poplinski, J. D., Koch, P. N., & Allen, J. K. (2001). Metamodels for computer-based engineering design: survey and recommendations. Engineering with computers, 17(2), 129-150.

[73] Smith, M. (1993). Neural networks for statistical modeling. Thomson Learning.

[74] Leary, S. J., Bhaskar, A., & Keane, A. J. (2004). A derivative based surrogate model for approximating and optimizing the output of an expensive computer simulation. Journal of Global Optimization, 30(1), 39-58.

[75] Sobol, I. M. (1998). On quasi-monte carlo integrations. Mathematics and Computers in Simulation, 47(2), 103-112.

[76] Swisher, J. R., Jacobson, S. H., & Yücesan, E. (2003). Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. ACM Transactions on Modeling and Computer Simulation (TOMACS), 13(2), 134-154.

[77] Thompson, S. K., & Collins, L. M. (2002). Adaptive sampling in research on risk-related behaviors. Drug and Alcohol Dependence, 68, 57-67.

[78] Thompson, S. K., & Seber, G. A. F. (1996). Adaptive sampling (p. 288). New York: Wiley.

[79] Truong, T. H., & Azadivar, F. (2003, December). Simulation optimization in manufacturing analysis: simulation based optimization for supply chain configuration design. In Proceedings of the 35th conference on Winter simulation: driving innovation (pp. 1268-1275). Winter Simulation Conference.

[80] Turner, C. J., Crawford, R. H., & Campbell, M. I. (2007). Multidimensional sequential sampling for NURBs-based metamodel development. Engineering with Computers, 23(3), 155-174.

[81] Williams, B. J., Santner, T. J., & Notz, W. I. (2000). Sequential design of computer experiments to minimize integrated response functions. Statistica Sinica, 10(4), 1133-1152.

[82] Xiong, F., Xiong, Y., Chen, W., & Yang, S. (2009). Optimizing Latin hypercube design for sequential sampling of computer experiments. Engineering Optimization, 41(8), 793-810

Biographical Information

John F Dickson was born in southern part of India where he earned his bachelor's degree in mechanical engineering from Noorul Islam College of engineering, Nagercoil, India. In 2007 he came to the US for his Master's program in Engineering Management at the University of Texas at Arlington. During his Master's program he got interested into statistics and continued his PhD at The Department of Industrial and Manufacturing Systems Engineering, University of Texas at Arlington in 2009. During his course of stay at UTA he worked in several funded research projects under Dr. John Priest, Dr. Susan Ferreira and Dr. Victoria Chen. He also worked as a teaching assistant at the Department of Industrial and Manufacturing Systems Engineering. His research interests are statistical modeling and optimization.