

# A Framework for Determination of the Weak Pareto Frontier Design Solutions under Probabilistic Constraints

Hongjun Ran\* and Dimitri Mavris†

*Aerospace Systems Design Laboratory, School of Aerospace Engineering  
Georgia Institute of Technology, Atlanta, GA 30332*

The design of complex systems such as aircraft or missiles requires the synergy of multiple disciplines. The design quality must ultimately be assessed by multiple criteria that often can not be optimized simultaneously. Therefore, in a less restrictive sense the Weak Pareto Frontier (WPF) in the objective space and the corresponding design solutions must be found because the WPF includes more compromised solutions than the conventional Pareto frontier. Real-world decisions are usually made in a state of uncertainty. Most often the effects of uncertainties are embodied in the probabilistic constraints (PC) that usually must be satisfied jointly. The combination of these issues requires a new framework to combine separately developing multidisciplinary optimization, multi-objective optimization, and joint probability assessment methods together, to solve a joint probabilistic constraint, multi-objective, multidisciplinary optimization problem and find the WPF solutions. The purpose of this paper is to provide such a framework. This framework starts with constructing fast and accurate surrogate models of different disciplinary analyses in order to reduce the computational time and expense to a manageable level and obtain trustworthy probabilities of the PC's and the WPF. A hybrid method is formed here that consists of the second order response surface methodology (RSM) and the support vector regression method (SVR) capturing the global tendency and local nonlinear behavior, respectively. The parameters of SVR to be pre-specified are selected using practical methods and a modified information criterion that makes use of model fitting error, predicting error, and model complexity information. Then a neighborhood search method based on Monte Carlo simulation is provided to find valid designs that are feasible and consistent for the coupling variables featured in a multidisciplinary design problem. Two schemes have been developed. One scheme finds the WPF by finding a large enough number of valid design solutions such that some WPF solutions are included in them. Another scheme finds the WPF by directly finding the WPF of each consistent design zone that is made up of consistent design solutions. Then the probabilities of the PC's are estimated, and the WPF and corresponding design solutions are found. A simple yet typical aircraft design problem is solved to demonstrate the feasibility of this framework. The results show that the method to select the pre-specified parameters of SVR works well, the hybrid surrogate models are fast and accurate, and both neighborhood search schemes can find the WPF.

## Nomenclature

$F$	=	vector of objective functions, or the cumulative distribution function
$F^{-1}$	=	inverse function of the cumulative distribution function
$g_j$	=	$j^{\text{th}}$ inequality limit state function
$g(\cdot)$	=	the general form of a surrogate model
$k(\cdot, \cdot)$	=	kernel function
$L_\varepsilon$	=	$\varepsilon$ -insensitive loss function
$n$	=	number of design variables

\* Graduate Research Assistant, Aerospace Systems Design Laboratory (ASDL), Student Member AIAA.

† Boeing Chair in Advanced Aerospace Systems Analysis, Director, ASDL, Associate Fellow AIAA.

$N$	=	required sample size of Monte Carlo simulation
$P(\cdot)$	=	probability
$P_f$	=	failure probability
$s$	=	size of a sample
$S$	=	sample, including values of both design variables and responses
$U$	=	uniform distributed random variable
$W$	=	weight vector of support vector regression model
$x_i$	=	$i^{\text{th}}$ design variable
$X$	=	vector of design variables, or a random variable
$y_i$	=	$i^{\text{th}}$ response or objective
$\hat{y}$	=	predicted value of the response
$\tilde{y}$	=	pseudo predicted value of the response by the $k$ -nearest-neighbor method
$\bar{y}$	=	mean of the response values of the sample
$Z$	=	value of a criterion
$\varepsilon$	=	tolerable deviation of support vector regression, or error of estimated probability
$\mu_i$	=	$i^{\text{th}}$ mean value of a random variable
$\sigma$	=	measure of errors, width of the radial basis function
$\Phi$	=	vector of pre-selected functions of support vector regression model
$\langle \cdot, \cdot \rangle$	=	dot product
Superscripts:		
$l$	=	lower bound
$u$	=	upper bound

## I. Introduction

Engineering design can be divided into three major phases: conceptual design, preliminary design, and detailed design.<sup>1</sup> Conceptual design can be further divided into requirement and function analysis, concept generation, concept selection, and design alternative selection. A design concept is an idea that represents a family of similar design alternatives and eventually is described by a parametric model with some design variables to predict or estimate the performance, quality, and cost of this family of design alternatives, while a design alternative is a specific design resulting from specific values of the design variables.<sup>2</sup>

With few exceptions the conceptual design of complex systems such as aircraft has many common features: multidisciplinary, multi-objective, design decisions are made under uncertainties, and decisions are made in a relatively short time period. These features make the design of new systems very complicated and difficult, and a different method is required for each of them.

The design of complex systems requires the synergy of multiple disciplines through the execution of complex computer analyses. These disciplines interact with each other through the coupling variables among them. A coupling variable is both one output of a disciplinary analysis and an input of another where the first disciplinary analysis directly or indirectly needs input information from the second one. A single discipline design usually does not have coupling variables because the outputs are seldom required to be fed back.

Coupling variables make multidisciplinary design much more difficult than single disciplinary design. There are three main impacts of these variables. First, design freedom is reduced because only some design points or solutions in the system level design space lead to converged values for the coupling variables. These points are called consistent design points, and they form disjointed zones in the design space, which are called consistent design zones. Second, they require many iterations of multidisciplinary analysis in order to find every consistent design point because of the disjointed consistent design zones. In other words, they entail equality constraints in the multidisciplinary analysis process complicating the design problem. Third, special solving procedures are required to decompose the complex interactions introduced by coupling variables and find consistent design points.

Multidisciplinary optimization (MDO) methods are formulated to integrate different disciplinary analyses together to handle the problems caused by coupling variables and enable a concurrent engineering process to solve such design and optimization problems efficiently. Depending on the availability of the optimization ability in the disciplinary contributing analyses (CA), different approaches are developed to find the optimal solutions from the consistent design zones:

- 1) If there is no optimization ability in the disciplinary CA's, the common approaches are the optimization with (relaxed) Fixed Point Iteration method (FPI) and the optimizer based decomposition method (OBD);
- 2) If all the disciplinary optimization ability is turned off and combined into a single system-level optimizer, the approach used is called the All-at-Once method (AAO);
- 3) If the disciplinary optimization ability is to be kept (only part of it in fact), multi-level MDO methods are used, such as Collaborative Optimization method (CO),<sup>3</sup> Modified Collaborative Optimization method (MCO),<sup>4</sup> and Bi-Level Integrated System Synthesis method (BLISS).<sup>5</sup>

Generally speaking, there are many potential design solutions to a multi-objective optimization problem (MOO). During the early stages of decision-making, one often has little information about the relative importance of the individual objectives. In these cases, a whole family of such solutions of which no objective can be further improved without degrading any other objectives must be considered with each solution being of equal importance. These solutions are denoted as Pareto frontier (PF) solutions. The purpose of mathematical optimization is therefore to give a variety of PF solution alternatives to the user (also called the decision-maker), or in the ideal case, to determine the entire set of PF solutions. However, the user may relax the requirements to accept such design solutions of which at least one objective is better than that of other design solutions, in order to have more trade-off choices. These relaxed MOO solutions are denoted as weak PF (WPF) solutions.

It can be shown that the methods to obtain PF solutions can easily be modified to obtain WPF solutions. Therefore, no difference is made for the methods to obtain PF and WPF solutions. Many approaches are developed to find a single PF solution. These approaches can be denoted as single solution MOO methods, such as the compromise programming method, the weighted sum method, and the goal attainment method.<sup>6</sup> However, special procedures are required to find a set of PF solutions, since it is desirable that such a method has practical attributes: a) it should generate evenly distributed PF points in the objective space; b) it should explore the entire objective space and not neglect any region.<sup>7</sup> To generate a set of evenly distributed PF points of the whole objective space, one approach is to systematically change the parameters in a single solution MOO method, such as the weights in the goal attainment method. However, this approach does not always result in an even distribution of PF points even though the weights are evenly varied.<sup>7</sup> Another approach is to use a MOO method that can directly generate such a set of PF points, such as the modified normal constraint (NC) method.<sup>8</sup>

Additionally, conceptual design decisions such as design alternative selection are made under uncertainties. Simply speaking, the source of uncertainty is the incompleteness of design knowledge. In more detail, uncertainties are caused by ambiguity of the requirements, variations in material properties, incomplete knowledge of the manufacturing process and operational environment such as variations in manufacturing precision and loading conditions, modeling assumptions, and other sources.

Uncertainties can and should significantly affect the decision-making process. Traditional multidisciplinary design optimization methods use a deterministic approach so that the optimal design solutions are frequently pushed to the limits of design constraint boundaries, leaving little or no room to accommodate uncertainties in system input, modeling, simulation, and operation environment.<sup>9</sup> As a result, these design solutions may be highly sensitive to the variation of the uncertain factors. This can lead to serious performance loss suffering from the high likelihood of undesired events such as off-design conditions, or being conservative and consequently uneconomic.

Most often the effects of uncertainties are embodied in the probabilistic constraints (PC), i.e. the probabilities of satisfying some constraints called limit state functions (LSF), must be greater than a prescribed level. For example, to consider the variation in material properties and/or load conditions, the PC can be stated as the probability that the maximum stress in a structure is less than a given level must be greater than 99.9%, or the failure probability less than 0.1%; to consider requirements subject to future changes, the designer can use a stricter than current requirement and the PC can be stated as the probability that this stricter requirement is satisfied must be greater than 85%. In most cases, the PC's are not independent because they share some random variables, and thus they must be satisfied jointly.

The key to considering uncertainties in the decision-making process is the ability to estimate the probability of satisfying limit state functions in the probabilistic constraints given uncertainties in design variables. The uncertainty in a design variable can be characterized in many ways: through probabilistic distributions for a random variable; by a range of expected variation in terms of low and high values; or by a delta or percent variation around a mean value.<sup>10</sup> Then joint probabilistic assessment (JPA) methods are used to estimate the probability of satisfying the LSF's. These assessment methods can be divided into two groups: simulation based and analytical. The most popular and most widely used simulation based method is the empirical distribution function (EDF) method based on Monte Carlo simulation. The most popular analytical methods are the fast probability integration (FPI) family of methods based on the concepts of most probable point (MPP).<sup>11</sup>

The above three features require the implementation of joint probabilistic (constraint), multi-objective, multidisciplinary optimization (JMOMDO) and finding the WPF solutions, in order to solve a realistic conceptual design problem with fewer simplifications than those made for problems with each of them. The observation on current practice shows that this kind of implementation has not yet been done, although some simpler implementations are performed, such as the component reliability assessment method<sup>12</sup> combining MDO and JPA; the fuzzy Pareto Frontier method<sup>13</sup> combining MDO, MOO, and finding PF; and the aero-propulsion component design method<sup>14</sup> combining MDO, MOO, and separate probabilistic assessment.

However, it is very difficult, if not impossible, to combine together current multi-disciplinary optimization methods, multi-objective optimization methods, and joint probability assessment methods to solve a JMOMDO problem and find the WPF solutions, since they are developed separately. For example, the MDO methods usually are developed to solve a single objective problem and they do not work well with MOO methods, and vice versa; the analytical probabilistic assessment methods do not perform well with complex multidisciplinary models as shown by experience; the Monte Carlo probabilistic assessment method can work well with complex models, but the computational load will be huge if it is directly combined with a MDO or MOO method. Therefore, a new framework is needed to overcome this obstacle.

As the last feature, conceptual design decisions are made in a relatively short time period, and this requires the design space to be explored quickly. Considering the vast computational load required to solve a JMOMDO problem and find the WPF solutions, it is impractical to use the time consuming complex disciplinary analyses in the design process. Instead, surrogate models have to be used in order to reduce the computational time and expense to a manageable level. On the other hand, since the conceptual design decisions have very important effects on final performance, quality, and 70 to 80 percent of the cost,<sup>15</sup> these surrogate models must also be accurate enough to obtain trustworthy probabilities of the PC's and WPF.

In this paper a new framework is introduced to solve a JMOMDO problem and find the WPF solutions for the conceptual design process. The remainder of this paper is organized as follows: Section II presents the mathematical and conceptual preliminaries; Section III presents the new framework; Section IV presents a simple yet typical aircraft design example; Section V gives the concluding remarks.

## II. Background

This section presents requisite mathematical and conceptual preliminaries, including the mathematical model of a JMOMDO problem, the concepts of PF and WPF, information criteria, response surface methodology (RSM), support vector regression (SVR), Monte Carlo simulation, and probabilistic assessment.

### A. The JMOMDO Problem Statement

A JMOMDO problem can be represented by the following mathematical model:

$$\text{Minimize: } F(X) = [f_1(X), f_2(X), \dots, f_e(X)]^T \quad \text{objective functions} \quad (1)$$

Subject to:

$$P(g_j(X) \leq th_j) \geq \alpha_j \quad j = 1, m \quad \text{(explicit) inequality PC's} \quad (2)$$

$$P(h_k(X) = th_k) \geq \beta_k \quad k = 1, l \quad \text{(explicit) equality PC's} \quad (3)$$

$$CX_q^I = CX_q^O \quad q = 1, c \quad \text{(implicit) coupling variable constraints} \quad (4)$$

$$X_i^l \leq X_i \leq X_i^u \quad i = 1, n \quad \text{side constraints} \quad (5)$$

where

$$X = [x_1, x_2, \dots, x_n]^T \quad \text{(random) design variables}$$

$$y_1 = f_1(X), y_2 = f_2(X), \dots, y_e = f_e(X) \quad \text{responses/objectives}$$

$$CX_q^I, CX_q^O \quad \text{input and output coupling variables}$$

$$g_j(X), h_k(X) \quad \text{limit state functions}$$

$th_j, th_k$ 

thresholds

 $\alpha_j, \beta_k$ 

required probabilities

The constraints  $g_j(X) \leq th_j$  and  $h_k(X) = th_k$  are deterministic constraints. The functions in Equations (1), (2), and (3) may be linear or non-linear functions. They may be explicit or implicit functions of design variables  $X$ . They may be evaluated by analytical or numerical techniques depending on if they are explicit functions. For most engineering problems, they are non-linear and implicit functions and have to be evaluated numerically using complex evaluation techniques such as finite element method.

The values of the design variables  $X$  are actually mean values, and for each there is a probabilistic distribution. So the objective of this design is to find the set of mean values of the design variables that satisfies the probabilistic constraints as well as minimizes the objective functions by adjusting the mean values of the design variables.

### B. The Concepts of PF and WPF

The mathematical definitions of the Pareto frontier and weak Pareto frontier points are as follows:<sup>16</sup>

Pareto frontier point: A point  $X^*$  is a Pareto frontier point if and only if there does not exist another point  $X^+$ , such that  $F(X^+) \leq F(X^*)$ , and  $F_i(X^+) < F_i(X^*)$  for at least one objective function, where  $F_i(X) = f_i(X)$  (see Equation (1)).

Weak Pareto frontier point: A point  $X^*$  is a weak Pareto frontier point if and only if there does not exist another point  $X^+$ , such that  $F(X^+) < F(X^*)$ .

According to the above definitions, a point is a WPF point if there is not another point that improves **all** of the objective functions simultaneously, while a point is a PF point if there is not another point that improves at least one objective function without degrading another objective function. Therefore, all PF points are WPF points, but not all WPF points are PF points. Figure 1 shows the difference between PF and WPF points. In this figure, the PF points are points b, e, and g, while the WPF points are points a, b, c, d, e, f, g, and h.

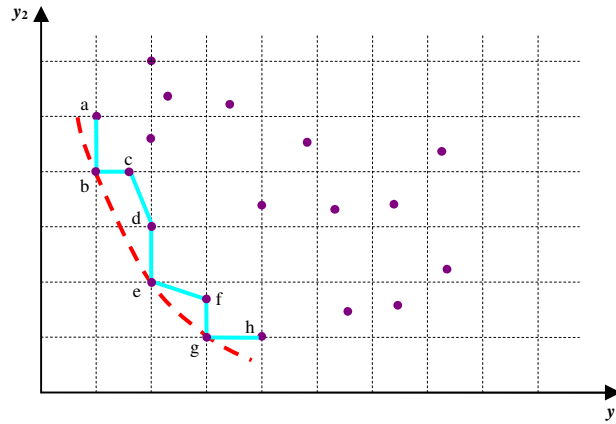


Figure 1. The PF and WPF in the Objective Space

### C. Information Criteria

Information criteria (ICs) are a class of methods that are developed specifically for the purpose of surrogate model selection and have been successfully used in many problems such as time series model selection. There are two popular information criterion methods: Akaike information criterion (AIC), and Bayesian information criterion (BIC). They ingeniously incorporate two information sources: the goodness of model fit and the complexity of a model, and reflects a balance between these two. The goodness of model fit is measured by the model fitting error, and the complexity of the model is measured by the number of parameters of the surrogate model. The practical forms of AIC and BIC are given as follows:<sup>17,18</sup>

$$AIC = \ln(\hat{\sigma}_{MLE}^2) + \frac{2m}{s} \quad (6)$$

$$\text{BIC} = \ln(\hat{\sigma}_{\text{MLE}}^2) + \frac{m \ln(s)}{s} \quad (7)$$

$$\hat{\sigma}_{\text{MLE}}^2 = \frac{\sum_{i=1}^s (y_i - \hat{y}_i)^2}{s} \quad (8)$$

where  $m$  is the number of parameters in the surrogate model,  $s$  is the size of the sample used for surrogate model construction,  $y_i$  is the true response value,  $\hat{y}_i$  is the predicted response value with the surrogate model, and  $\hat{\sigma}_{\text{MLE}}^2$  is the squared maximum likelihood estimation of the model fitting error.

According to the above equations, the main difference between AIC and BIC is that BIC puts a greater penalty on the model complexity when the sample size is larger than 8. Therefore, BIC will select a model with a number of parameters no greater than that chosen by AIC.

#### D. Response Surface Methodology

Response surface Methodology (RSM) is a well investigated and commonly applied surrogate-modeling method in engineering designs.<sup>19</sup> It has been successfully used for a variety of applications in different areas. RSM uses polynomials to approximate the true response behavior, and the polynomials are called a response surface equation (RSE). Usually a second order polynomial equation is used, the main reason being that considerable practical experience has shown that a second order model works well for many real problems. Higher order terms can be added in if needed. The general second order RSE including linear, quadratic and interaction terms is as follows:

$$\hat{y} = b_0 + \sum_{i=1}^n b_i x_i + \sum_{i=1}^n b_{ii} x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^n b_{ij} x_i x_j \quad (9)$$

where  $x_i$  are the design variables,  $n$  is the number of design variables, and  $(b_0, b_i, b_{ii}, b_{ij})$  are the parameters to be estimated from the sample data. The total number of parameters are  $(n+1)(n+2)/2$ . Usually the parameters are estimated by the least squares method.

#### E. Support Vector Regression

Support vector regression is a state-of-the-art surrogate modeling method developed from the support vector machine (SVM). Its main advantages include: good accuracy for most types of problems; no problem of ‘‘curse of dimensionality’’ that troubles many existing surrogate modeling methods; the final surrogate model is unique because the optimization problem for model fitting has only one extreme solution, the global optimal solution; the complexity of the final surrogate model is controllable, i.e. not all the sample data are needed in the final model, while all other no-curse-of-dimensionality methods such as neural networks require the complete set of sample data in the final model; last but not the least, fast model fitting speed. The main reason this method is not popular yet is that this method requires the user to pre-specify three parameters before the model fitting optimization process.

For SVR, the regression/surrogate model is assumed to have the following form:<sup>20</sup>

$$\hat{y} = g(X) = \langle W, \Phi(X) \rangle + b \quad (10)$$

where  $\langle \cdot, \cdot \rangle$  means dot product,  $X$  is the vector of design variables,  $W$  is a vector of scalars (weights) to be estimated,  $b$  is the bias or intercept to be estimated, and  $\Phi(X)$  is a pre-selected function vector of which the form does not need to know yet. The size of vector  $W$  is equal to the size of the sample data  $s$ . The vector  $W$  and  $b$  are the parameters to be estimated from the sample data, and the total number of parameters are  $(s+1)$ .

The above parameters are estimated in the following way. First a loss function is selected. The  $\varepsilon$ -insensitive loss function is the most popular because of its many good characteristics, shown as follows:

$$L_\varepsilon(X, y, g(X)) = L_\varepsilon(y - g(X)) = \begin{cases} 0, & \text{if } |y - g(X)| < \varepsilon \\ |y - g(X)| - \varepsilon, & \text{otherwise} \end{cases} \quad (11)$$

Then a practical structural risk function is constructed that is based on the loss function, shown as follows:

$$R_s = \frac{1}{s} \sum_{i=1}^s L(X_i, y_i, g(X_i)) + \frac{1}{2} \|W\|^2 \quad (12)$$

Then the parameters are estimated by minimizing the practical structural risk function. This optimization problem is converted to an alternative convex quadratic minimization problem that has only one extreme solution, the global optimal solution, shown as follows:

$$\begin{aligned} & \underset{W, b, \xi_i^+, \xi_i^-}{\text{Minimize}} \quad \frac{1}{2} \|W\|^2 + C \sum_{i=1}^s (L(\xi_i^- + \varepsilon) + L(\xi_i^+ + \varepsilon)) \\ & \text{subject to} \quad \begin{cases} y_i - \langle W, \Phi(X_i) \rangle - b \leq \varepsilon + \xi_i^+ \\ \langle W, \Phi(X_i) \rangle + b - y_i \leq \varepsilon + \xi_i^- \\ \xi_i^+, \xi_i^- \geq 0 \end{cases} \end{aligned} \quad (13)$$

The solution of the above optimization problem has the following form:

$$\hat{y} = g(X) = \langle \bar{W}, \Phi(X) \rangle + \bar{b} = \sum_{i=1}^s (\alpha_i^+ - \alpha_i^-) k(X_i, X) + \bar{b} \quad (14)$$

where  $\alpha_i^+$  and  $\alpha_i^-$  are dual variables in the Lagrangian function constructed from the optimization problem (13),  $k(X, X')$  is called the kernel function,  $\bar{b}$  is the estimation of  $b$ ,  $C$  is called the regularization factor,  $\varepsilon$  is called the tolerable deviation.

There are many kernel functions  $k(X, X')$ . One of them is the Gaussian radial basis function (GRBF) that has good performance for all kinds of problem, shown as follows:

$$k(X, X') = \exp\left(-\frac{\|X, X'\|^2}{2\sigma^2}\right) \quad (15)$$

where  $\sigma$  is called the width of the radial basis function.

$C$ ,  $\varepsilon$ , and  $\sigma$  are the three parameters that must be pre-specified before the model fitting optimization process. Their values will greatly affect the performance of the final surrogate model in terms of accuracy and complexity. Therefore, their values have to be carefully specified. This work used to be done by the SVR experts, and this is the main reason SVR is not popular yet. In this paper, practical methods are proposed to specify the values of these three parameters, and thus a non-expert user can use SVR to obtain good SVR surrogate models.

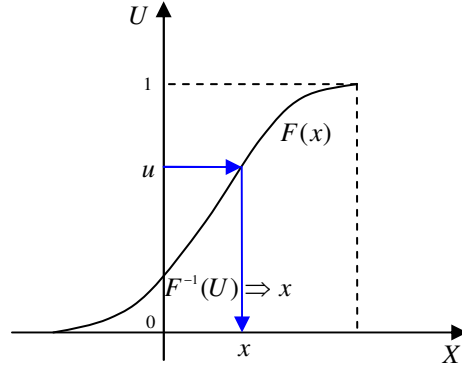
## F. Monte Carlo Simulation

Monte Carlo simulation (MCS) is a technique to simulate a random process given the probabilistic distributions of the random variables. The basic idea of MCS is as following: first a set of sample points of the random variables are generated by the Monte Carlo sampling method; then the responses of the sample points are obtained. Thus a random sample is obtained or a random process is simulated.

Here is the process of the univariate Monte Carlo sampling. Suppose  $F(x)$  is the cumulative distribution function (CDF) of a random variable  $X$ , then the MC sampling method generates the sampling points as

$$x = F^{-1}(U) \quad (16)$$

where  $F^{-1}(\cdot)$  is the inverse function of  $F(x)$ , and  $U$  is a uniform random variable of which values are generated by a (pseudo-) random number generator in computer experiments. Figure 2 shows the process to generate a sample point by MC sampling.



**Figure 2. The Univariate Monte Carlo Sampling Process**

If the number of random variables is more than one, the multi-variate Monte Carlo sampling method has to be used. However, for most engineering design problems the random variables are independent, and thus the above univariate Monte Carlo sampling method is used to generate the sample.

### G. Joint Probabilistic Assessment

Here only the MCS based joint probabilistic assessment method is discussed. This method is also called the empirical distribution function (EDF) method. First the sample is generated by MCS. Then the joint CDF of the sample is generated using the following counting equation:

$$F(y_1 \leq a_{i1}, y_2 \leq a_{i2}, \dots, y_m \leq a_{im}) = \frac{1}{N} \sum_{i=1}^N I(y_1 \leq a_{i1}, y_2 \leq a_{i2}, \dots, y_m \leq a_{im}) \quad (17)$$

where  $N$  is the number of sample points,  $Y_1, Y_2, \dots, Y_m$  are response functions of random variables  $X_1, X_2, \dots, X_n$ ,  $a_i$  are pre-specified values for the response functions, and  $I(\cdot)$  is the indicator function, giving 1 if the conditions in the parenthesis are all satisfied and 0 otherwise.

In general, the joint CDF of the sample is different from that of the random process. However, in practice the joint counting equation (17) is directly used to calculate the probability of satisfying the probabilistic constraints jointly.

## III. Proposed Methods

This section presents the new framework to solve a JMOMDO problem and find the WPF solutions for the conceptual design process. This framework starts with constructing fast and accurate surrogate models of different disciplinary analyses. A new surrogate modeling method, a hybrid method of RSM and SVR, is proposed. Then a new neighborhood search method based on Monte Carlo simulation is provided to find consistent designs featured by a multidisciplinary design problem. Two schemes are developed for this search method. At last, the overall flowchart of this framework is provided.

### A. Hybrid Surrogate Modeling of RSM and SVR

As mentioned before, fast and accurate surrogate models of different disciplinary analyses are needed in order to reduce computational time and expense and obtain trustworthy probabilities of the PC's and WPF. Although SVR can obtain good accuracy for most problems, for some problems it still lacks the required accuracy. For these problems, a hybrid combination of RSM and SVR is found to provide better accuracy than either RSM or SVR. This is because RSM can capture the global tendency very well, while SVR can capture local nonlinear behavior very well. This treatment is analogous to the process of dividing a complex job into smaller and simpler parts, finding the best method for each part, and finally obtaining the best result. This hybrid surrogate modeling method can rapidly provide fast and accurate surrogate models for all kinds of engineering problems.

The three parameters of SVR to be pre-specified are selected using practical methods and a modified information criterion that makes use of model fitting error, predicting error, and model complexity information. This selection process is automatic, and thus all users can take advantage of SVR, a state-of-the-art surrogate modeling method.



### 1. The Modified Information Criteria

As mentioned before, the current practical forms of AIC and BIC criteria make use of model fitting error and model complexity. A good surrogate model should have low model fitting error (MFE), complexity, and model prediction error (MPE). One problem with these criteria is that the model prediction error of the surrogate model selected by them may not be low, because low MFE can not guarantee low MPE. Therefore, the authors suggest including the model predicting error in the AIC and BIC so as to select a surrogate model that is balanced between MFE and MPE. Such modified AIC and BIC are denoted as AICC and BICC.

One popular model error measure is the root mean square error (RMSE), given as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n_{\text{error}}} (y_i - \hat{y}_i)^2}{n_{\text{error}}}} \quad (18)$$

where  $n_{\text{error}}$  is the number of points used to calculate the error, including sample points and/or new points.

It can be easily shown that  $\hat{\sigma}_{\text{MLE}}^2$  is squared RMSE of MFE, so it will be denoted as  $\text{RMSE}_{\text{MFE}}^2$  hereafter. Similarly,  $\text{RMSE}_{\text{MPE}}^2$  is used to represent the squared model predicting error. Then general forms of AICC and BICC are given as

$$\text{AICC} = \frac{1}{2} (\ln(\text{RMSE}_{\text{MFE}}^2) + \ln(\text{RMSE}_{\text{MPE}}^2)) + \frac{2m}{s} \quad (19)$$

$$\text{BICC} = \frac{1}{2} (\ln(\text{RMSE}_{\text{MFE}}^2) + \ln(\text{RMSE}_{\text{MPE}}^2)) + \frac{m \ln(s)}{s} \quad (20)$$

The MFE can be easily estimated with Equation (18) based directly on the sample data. If new randomly generated sample points are available, MPE can be easily estimated with Equation (18) based on this new sample and the fitted surrogate model, otherwise special methods have to be used. The popular methods to estimate MPE without new sample points are cross validation<sup>21</sup> and bootstrap.<sup>22</sup>

### 2. The Methods to Select the Three Parameters of SVR

Through the years, people have developed practical methods to select two pre-specified parameters of SVR, i.e.  $C$  and  $\varepsilon$  based on the information contained in the sample. These methods are given as follows:<sup>23</sup>

$$C = \max(|\bar{y} + 3\sigma_y|, |\bar{y} - 3\sigma_y|) \quad (21)$$

where  $\bar{y}$  and  $\sigma_y$  are the mean and the standard deviation of the response values of the sample, respectively.

$$\varepsilon = 3\hat{\sigma}_E \sqrt{\frac{\ln s}{s}}, \text{ or } \varepsilon = \frac{\hat{\sigma}_E}{s} \quad (22)$$

A  $k$ -nearest-neighbor method is used to estimate  $\hat{\sigma}_E$  as follows:

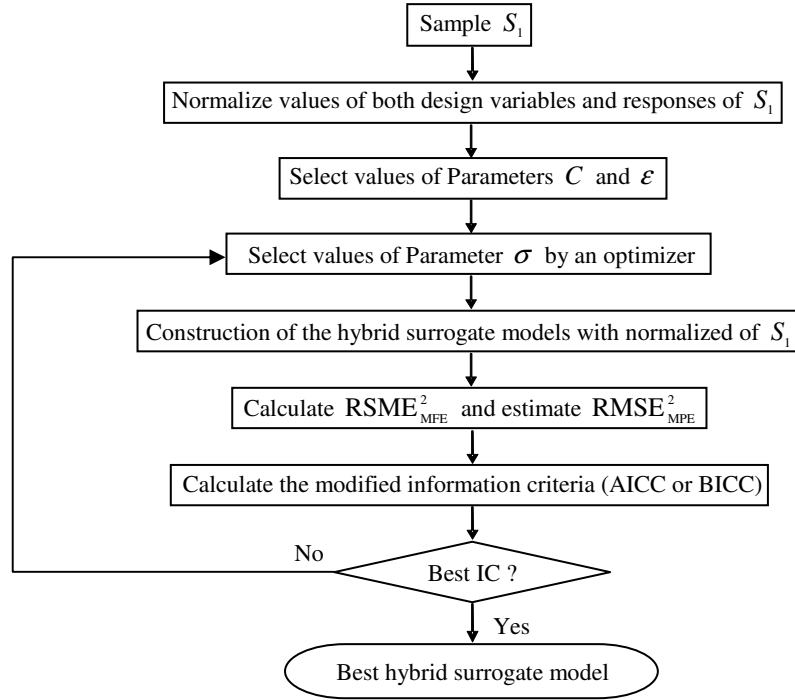
$$\hat{\sigma}_E^2 = \frac{s^{\frac{1}{2}} k}{s^{\frac{1}{2}} k - 1} \frac{1}{s} \sum_{i=1}^s (y_i - \tilde{y}_i)^2 \quad (23)$$

where  $\tilde{y}_i$  is the (pseudo) predicted response value of each sample point estimated as the average of the responses values of the  $k$  nearest sample points, where the distance between two sample points is measured by their Euclidian distance. Typically, the value of  $k$  is in the 2 – 6 range, and a value of 3 is recommended.

The last pre-specified parameter of SVR, i.e.  $\sigma$ , is selected by minimizing the modified information criterion (AICC or BICC), and the final surrogate model is determined at the same time this parameter is optimized. See the following flowchart in Figure 3.

### 3. The Flowchart for Hybrid Modeling of RSM and SVR

The basic idea of the hybrid modeling method of RSM and SVR is as follows: first the sample is fitted by RSM; then the residuals of the RSM model are fitted by SVR. In order to select the pre-specified parameter  $\sigma$  of SVR no matter the magnitude of values of the responses, both the design variable values and response vales are normalized. Figure 3 shows the flowchart for hybrid modeling method of RSM and SVR.



**Figure 3. The Flowchart of Hybrid Modeling of RSM and SVR**

## B. The Neighborhood Search Method

A new neighborhood search method based on Monte Carlo simulation is used to decompose the interactions among different discipline analyses and enable finding the WPF under PC's at the same time at short computational time and low cost.

### 1. The Basic Idea of the Neighborhood Search Method

Considering what is needed for a method to solve a JMOMDO problem and find WPF, this method should:

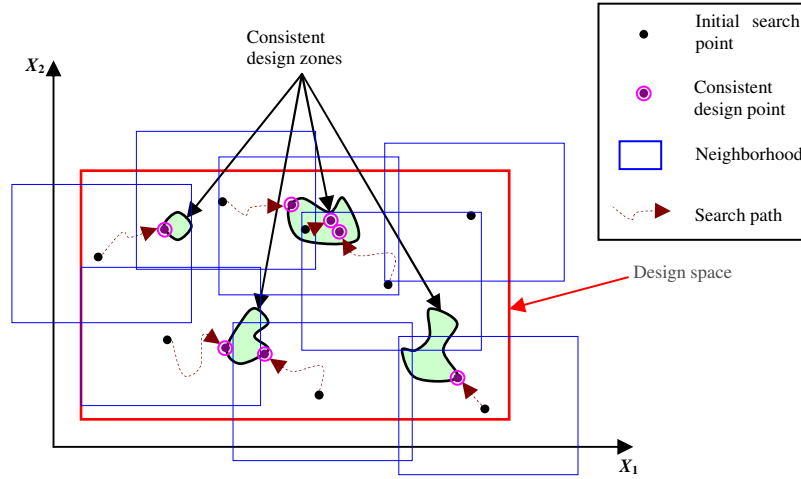
- 1) find a MDO solution under deterministic constraints (i.e. consistent and feasible solution, or denoted as valid solution) very fast and at low cost;
- 2) explore the entire design space without missing any disjointed consistent design zone;
- 3) on top of requirement 2), be able to find the WPF under deterministic constraints over each disjointed consistent design zone, i.e. local deterministic WPF, if any;
- 4) find enough and evenly distributed points for each local deterministic WPF;
- 5) on top of all above, find the global WPF under probabilistic constraints, i.e. global probabilistic WPF over the whole design space, if any;
- 6) find enough and evenly distributed points for the global probabilistic WPF.

Observing the current methods to combine together some of MDO, MOO, and JPA, usually they adopt a multi loop approach with one loop each to handle MDO, MOO, and JPA, respectively. Although this approach has been successfully applied in many problems and the computational time and cost are acceptable since there are at most two loops nested together, it will result in unacceptable long computational time and high cost if solving a JMOMDO problem and finding the WPF since there are three loops nested together and both computational time and cost increase exponentially with the number of loops.

One may consider using domain spanning search methods such as grid search and random search methods since they do not have the problem of nested loops. The grid search method establishes a grid network in the design space and uses the grid knots as search points. This method can guarantee a uniform distribution of the search points over the design space, but the difficulty comes from choosing the appropriate fineness of the grid. With a coarse grid network, one may not obtain any consistent designs in some consistent design zones. Another problem with a coarse grid network is that one may not obtain enough consistent design points to find the WPF with a certain confidence. On the other hand, with a fine grid network, the computational time may be unacceptable, especially for high dimensional design problems. The random search method generates random search (design) points and checks the convergence criteria for the coupling variables. There are two main problems with this random search method. First,

because the search points are randomly generated, it is possible that only a very few of them, if not none of them, are found to be consistent designs. Second, one does not know when to stop the search, because it is hard, if not impossible, to know if the randomly generated search points are uniformly distributed over the whole design space. For these reasons, the domain spanning search methods are not adopted.

However, combining the ideas of grid search and random search together and making modifications, a new neighborhood search method is formed to efficiently solve a JMOMDO problem and find WPF without nested loops. This approach starts with a set of initial search points that are generated over the design space by uniform Monte Carlo sampling. This set of sampling points is denoted as sampling points  $S_2$ . Then a hypercube is defined to be centered at each starting point. The hypercube of each starting point is called the neighborhood of this point. The neighborhoods should be defined such that they can cover the entire design space, although there may be overlapping among them. Then an optimizer is used to search for a consistent design point within this hypercube. Figure 4 illustrates the idea of this approach.



**Figure 4. Illustration of the Neighborhood Search Method**

As a rule of thumb, the length of  $i^{\text{th}}$  side of the hypercube for overlapping is given as

$$a_i = \frac{2}{\sqrt[n]{s_2}} \Delta x_i, i = 1, \dots, n \quad (24)$$

where  $a_i$  is the length of  $i^{\text{th}}$  side of the hypercube,  $\Delta x_i$  is the range of  $i^{\text{th}}$  design variable,  $s_2$  is the size of  $S_2$ .

Denoting an initial search point as  $X_0$ , then the lower and upper bounds of  $i^{\text{th}}$  design variable in a neighborhood search problem is given as

$$x_{i0} - \frac{a_i}{2} \leq x_i \leq x_{i0} + \frac{a_i}{2}, i = 1, \dots, n \quad (25)$$

This neighborhood search method has many advantages. First, unlike the modified normal constraint method<sup>8</sup>, no optimization effort is used to generate initial search points. Second, like the modified NC, this method also guarantees evenly distributed global PF points over the whole design space. This method combines the flexibility of the random search method and uniformity of the grid search method. Since this set of initial search points is generated over the design space by uniform Monte Carlo sampling, they are randomly distributed while covering the entire design space uniformly at the same time. As long as the neighborhood of a starting point overlaps with a consistent design zone, the optimizer usually can find a consistent design either on the boundary of this consistent design zone or inside it. Since the initial search points are uniformly distributed, under this mechanism no disjointed consistent design zone will be missed as long as the neighborhoods will cover the entire design space. Usually an optimizer will reach a different consistent design if starting from a different initial search point. Since the initial search points are randomly distributed, under this mechanism the consistent design solutions will be randomly distributed and some of them will be on or near the local deterministic WPF's. Since each neighborhood is small, it is almost impossible for a neighborhood to contain more than one local deterministic optimal design. Therefore, if the optimizer is used to directly search for a local deterministic WPF point of the part of a consistent design zone

within a neighborhood, it will almost surely find such a point. Therefore, the global (probabilistic) WPF can be found by either indirectly or directly searching for local deterministic WPF's, and the representing discrete points of the global WPF will be evenly distributed. From the above description one can see that this neighborhood search method solves a MDO problem in a way similar to the OBD method.

The third advantage of this method is that the required number of initial search points can be estimated since it is based on Monte Carlo simulation. If one thinks the instance of obtaining a global WPF is probabilistic, as a rule of thumb, the size of  $S_2$  can be estimated as

$$s_2 = 2n_{cx}s_0 \quad (26)$$

where  $n_{cx}$  is the number of coupling variables, and  $s_0$  is the  $N$  in the following equation:<sup>24</sup>

$$\varepsilon = \sqrt{\frac{200 \cdot (1 - P_f)}{N \cdot P_f}} \quad (27)$$

where  $N$  is the number of analyses or simulations required, and  $P_f$  is the failure probability (or the probability of not finding the global PF),  $\varepsilon$  (%) is the probability calculation error of the Monte Carlo simulation. As one example, if the required  $P_f = 1\%$  and  $\varepsilon = 2$  (%), then the required  $N = 4,500$ .

Undeniably, this neighborhood search method requires a large number of initial search points since it is based on Monte Carlo simulation. However, this large number of initial search points is necessary for finding a global WPF under probabilistic constraints. On the other hand, one may suspect this method requires unacceptable computational time since there are such a large number of neighborhoods to be searched by an optimizer. However, since each neighborhood is small and usually contains one local optimal design, the optimization process is fast in a neighborhood. Therefore, the overall computational time is not long and manageable.

## 2. The First Search Scheme

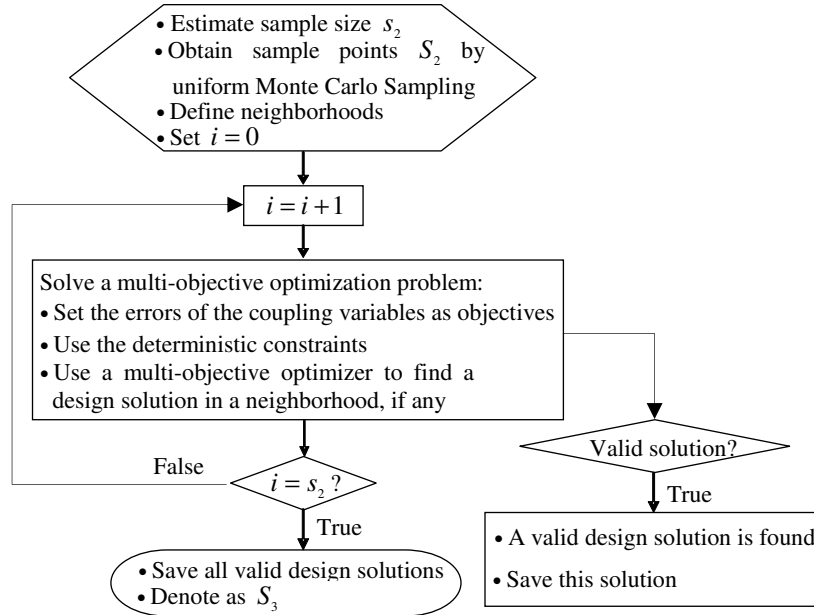
In the above general description of the neighborhood search method, an optimizer is used to search for a consistent design solution in a neighborhood. Obviously, a consistent design is not enough, because a consistent design may not be feasible, i.e. satisfying the (explicit) deterministic constraints. A consistent and feasible solution is denoted as a valid solution. As the first search scheme, the optimizer will search for a valid design solution in a neighborhood, if any exists. With this search scheme, the global WPF will be found by indirectly searching for local deterministic WPF's.

Figure 5 shows the first scheme. In this scheme, the multi-objective optimizer used is the goal attainment optimizer "fgoalattain" in Matlab<sup>®</sup>. For this optimizer, the goals are set as  $\text{Goals} = [0 \cdots 0]$ , and the weights are set as  $\text{Weights} = [1/e \cdots 1/e]$ . All the valid solutions obtained are denoted as  $S_3$ .

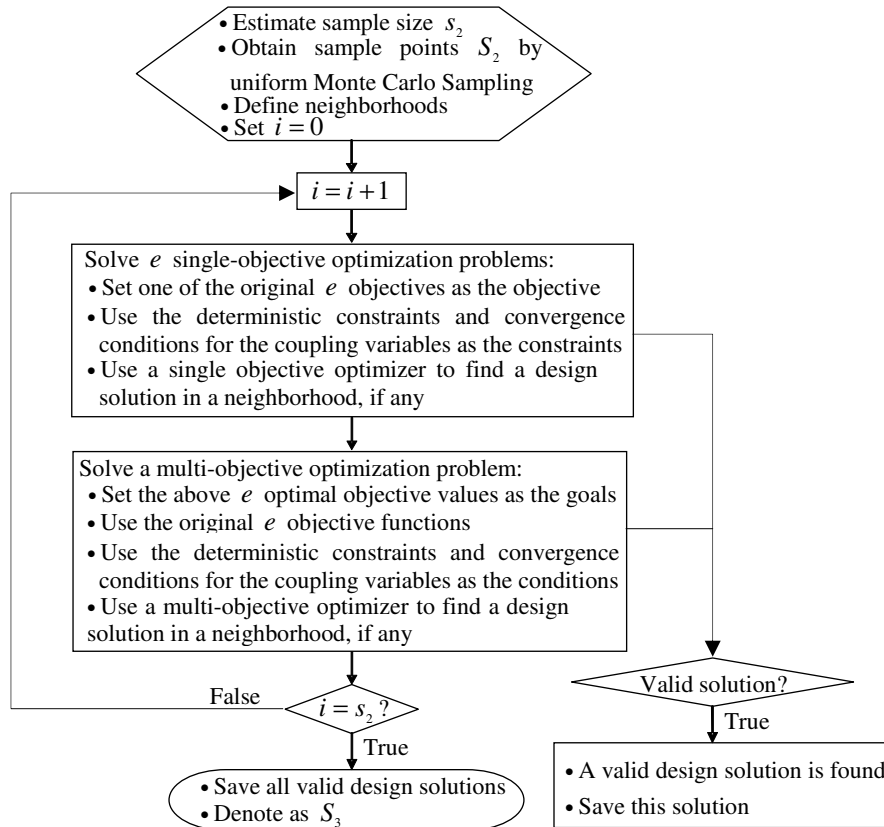
## 3. The Second Search Scheme

The second scheme directly searches for a local deterministic WPF point of the part of a consistent design zone within a neighborhood. This local deterministic WPF point may also be a local deterministic WPF point of the whole consistent design zone if part of the boundary of the whole consistent design zone is within the neighborhood. With this scheme, the global WPF will be found by directly searching for local deterministic WPF's.

Figure 6 shows the second scheme. This scheme first searches for  $e$  optimal valid design solutions of  $e$  single-objective optimization problems, if any are in a neighborhood. The objective function of each single-objective optimization problem is one of the  $e$  objectives of the original design problem. The constraints of each of the single-objective optimization problem include all the deterministic constraints and  $n_{cx}$  convergence conditions for the coupling variables. Then the above  $e$  objective function values are set as the goals for a multi-objective optimization problem. The objective functions of this multi-objective optimization problem are the  $e$  objectives of the original design problem, and the constraints are all the deterministic constraints and the convergence conditions for the coupling variables. Note that the optimal valid design solutions of the single-objective optimization problems are also deterministic WPF points. By this way, the deterministic WPF of the part of a consistent design zone in a neighborhood is represented by  $(e+1)$  points, if any in a neighborhood. The "fmincon" optimizer in Matlab<sup>®</sup> is used for the single-objective optimization problems, and the "fgoalattain" optimizer is used for the multi-objective optimization problem. For the multi-objective optimizer, the weights are set as  $\text{Weights} = [1/e \cdots 1/e]$ . Since the optimization problem of the second scheme is much more complicated than the first one, the second scheme is more computationally demanding and thus more time consuming.



**Figure 5. The First Scheme for the Neighborhood Search Method**



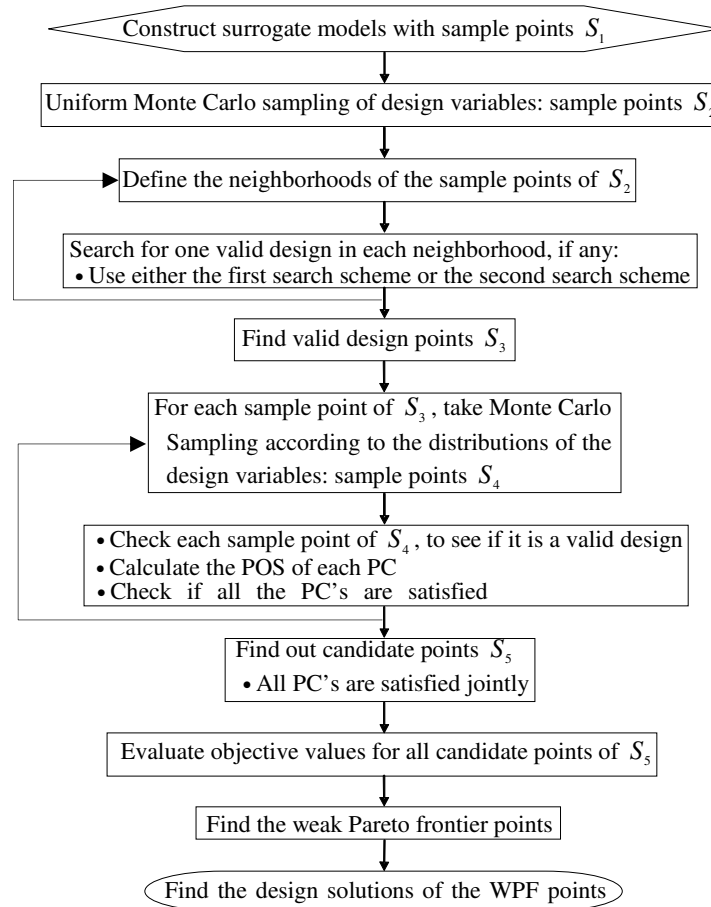
**Figure 6. The Second Scheme for the Neighborhood Search Method**

**C. The Framework**

After obtaining the valid design solutions  $S_3$ , for each design of  $S_3$ , the values of the design variables are treated as mean values, and a Monte Carlo sampling is executed according to the distributions of the design

variables about this point of  $S_3$ . The sample size estimation is given by Equation (27). The resulted sample points are denoted as  $S_4$ .

Each design point of  $S_4$  is checked to see if it is a valid design, and the probabilities of satisfying each PC (probability of success, POS) are calculated for all valid design points of  $S_4$  using Equation (17). Then probabilities of the valid design points of  $S_4$  are used to check if they satisfy the PC's jointly. If all the PC's are satisfied jointly, the corresponding design solution of  $S_3$  is saved as a candidate design. A candidate design solution is a design solution of  $S_3$  of which random sample points  $S_4$  result in satisfaction of all PC's jointly. The resulted candidate points are denoted as  $S_5$ . Then the multiple objective values of the candidate points  $S_5$  are evaluated, and are used to discover the points on the WPF in the objective space. As the last step, the design solutions of  $S_5$  corresponding to these WPF points are located accordingly, and are denoted as  $S_{WPF}$ . Figure 7 shows the overall flowchart of the framework.



**Figure 7. The Overall Flowchart of the Framework**

#### IV. A Simple Aircraft Design Example

A simple yet typical aircraft design problem is used to show the feasibility of the new framework for determination of the WPF solutions under probabilistic constraints. The design is performed with accurate surrogate models of discipline analyses. Both search schemes of the neighborhood search method are used to solve this problem, and the results are compared.

##### A. The Problem Setup

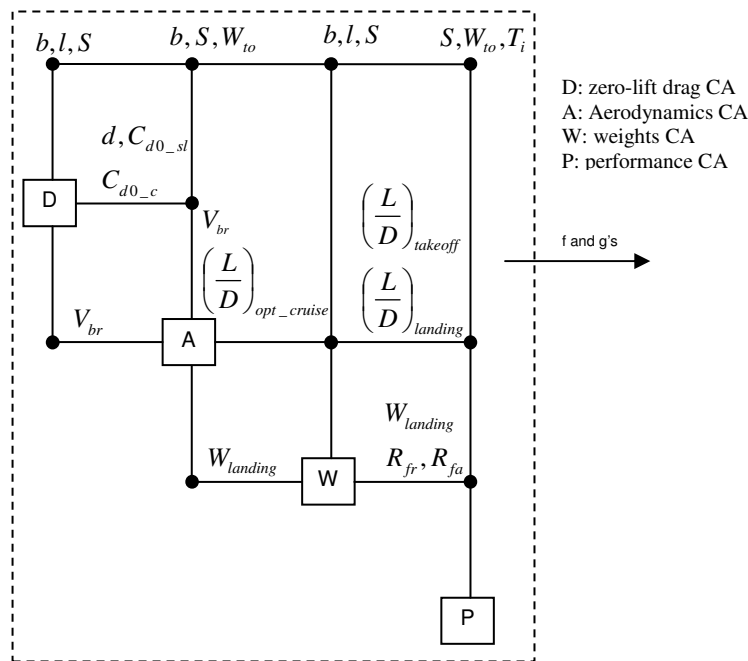
This aircraft design example is to design a mid-range passenger jet transport. There are two design objectives: productivity index (PI), to be maximized, and installed total engine thrust ( $T_i$ ), to be minimized. PI is a measure of

the speed and cargo carrying capability of an aircraft normalized by the sum of its empty and fuel weights.  $T_i$  is considered to be a measure of both purchase price and operational cost.

This design problem has 4 disciplinary contributing analyses (CA), shown in Figure 8. A design structure matrix is used to show how the disciplinary CA's are related, what the inputs and outputs of each CA are, etc. This problem has 5 system level design variables, and Table 1 shows their meaning, ranges of means, and distributions.

**Table 1. The System Level Design Variables of the Aircraft Design Example**

Variable	Name	Range of Mean	Distribution
$b$	wing span (ft)	[95 145]	normal, $\sigma = \text{range}/2000$
$l$	fuselage length (ft)	[120 140]	normal, $\sigma = \text{range}/2000$
$S$	wing area (ft <sup>2</sup> )	[1300 1850]	normal, $\sigma = \text{range}/2000$
$W_{to}$	takeoff gross weight (lb)	[155000 180000]	normal, $\sigma = \text{range}/2000$
$T_i$	installed total engine thrust (lb)	[20000 35000]	normal, $\sigma = \text{range}/2000$



**Figure 8. The Design Structure Matrix of the Aircraft Design Example**

According to the DSM, this design problem has two coupling variables:  $V_{br}$ , the best range cruise speed, and  $W_{landing}$ , the landing gross weight. This design problem has seven probabilistic constraints, shown in Table 2. The required probability of success for the all the PC's are at least 0.85.

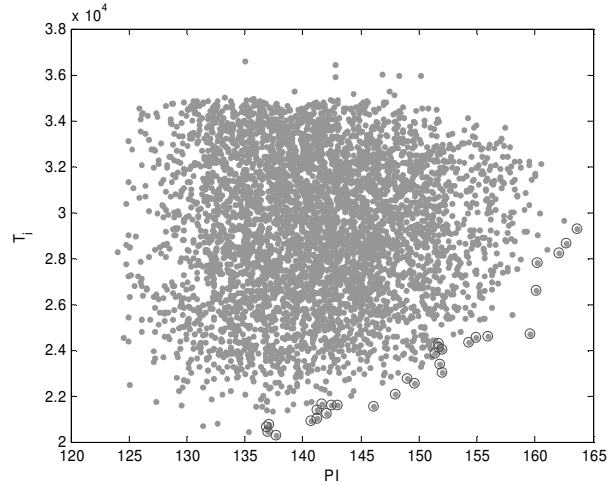
**Table 2. The Probabilistic Constraints of the Aircraft Design Example**

$P(S_{to} \leq 6000) \geq 0.85$	takeoff field length (ft)
$P(S_l \leq 4000) \geq 0.85$	landing field length (ft)
$P(U \geq 0.3) \geq 0.85$	useful load fraction
$P(q_{to} \geq 2.7\%) \geq 0.85$	takeoff climb gradient (one engine out)
$P(q_l \geq 2.4\%) \geq 0.85$	aborted landing climb gradient (one engine out)
$P(\bar{R}_f \geq 1) \geq 0.85$	overall mission fuel balance (available/required)
$P(AR \leq 10.5) \geq 0.85$	wing aspect ratio

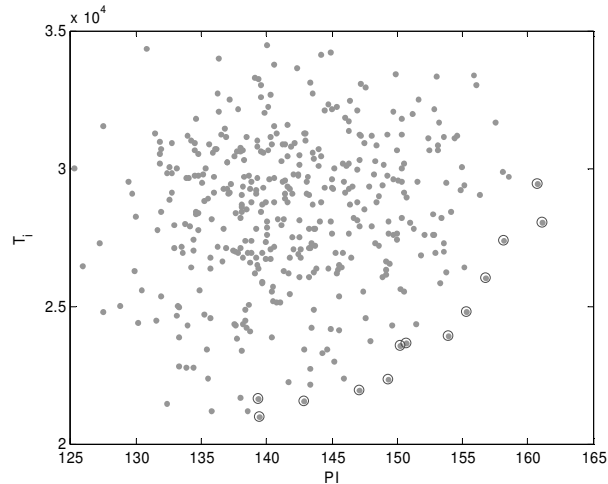
## B. Design Results

A surrogate model is constructed for each response or output of the disciplinary analyses. The accuracy of the surrogate models is satisfactory, since the maximum model predicting error is less than 3% of the range of a response. The solving process starts with 19,800 sample points of  $S_2$ . This number is estimated by Equation (26) and (27) with  $P_f = 1\%$  and  $\epsilon = 2$  (%).

With the first search scheme, 7422 valid solutions  $S_3$  are obtained; then 4601 candidate points of  $S_3$  are obtained; and finally 30 global WPF points are obtained. With the second search scheme, 10761 valid solutions  $S_3$  are obtained; then 410 candidate points of  $S_3$  are obtained; and finally 13 global WPF points are obtained. Figure 9 and Figure 10 show the global WPF's in the objective space found by the two search schemes, respectively. In these two figures, all the points are candidate points.



**Figure 9. The Global WPF Found by the First Search Scheme**



**Figure 10. The Global WPF Found by the Second Search Scheme**

Observing these two global WPF's in the objective space, one can find: first, they are similar, although the number of representing points are different; second, the representing points are quite evenly distributed, although the global WPF of the first search scheme has some clusters of points. Therefore, we can draw the conclusion that both search schemes of the neighborhood search method found the correct global WPF.

## V. Conclusion

A framework is proposed to solve realistic conceptual design problems of complex systems featuring complicated interactions among multiple disciplinary analyses, uncompromising tradeoff among multiple design objectives, and quick yet responsible decisions made under high uncertainties. The results of this framework are the



best compromised design solutions, i.e. global WPF that considers the effects of uncertainties so that the decision maker can make educated decisions. A simple yet typical aircraft design example demonstrates the feasibility or usefulness of this framework.

A hybrid surrogate modeling method of RSM and SVR is proposed and shown to be able to accurately simulate the complex and time-consuming disciplinary analyses such that reliable results are obtained and computational time and cost are reduced to a manageable level. A Monte Carlo simulation based neighborhood search method is proposed and shown to be able to efficiently solve complicated realistic conceptual design problems.

### Acknowledgments

The Hongjun Ran author can not be thankful enough to his advisor Dr. Mavris and his peers for their wise comments and help.

### References

- <sup>1</sup>Raymer, Daniel P., *Aircraft Design: A Conceptual Approach*, 3<sup>rd</sup> ed., AIAA, Virginia, 1999.
- <sup>2</sup>Mattson, C., Messac, Achille, "Pareto Frontier Based Concept Selection under Uncertainty, with Visualization", *Optimization and Engineering*, Vol. 6, 2005, pp. 85.
- <sup>3</sup>Kroo, I., Manning, V., "Collaborative Optimization: Status and Directions", 2000, AIAA-2000-4721.
- <sup>4</sup>DeMiguel, A.-V., Murray, Walter, "An Analysis of Collaborative Optimization Methods", 2000, AIAA-2000-4720.
- <sup>5</sup>Sobieszczanski-Sobieski, J., Altus, Troy D., et al, "Bi-Level Integrated System Synthesis (BLISS) for Concurrent and Distributed Processing", 2002, AIAA-2002-5409.
- <sup>6</sup>Marler, R. T., Arora, J. S., "Survey of Multi-Objective Optimization Methods for Engineering", *Structural and Multidisciplinary Optimization*, Vol. 26, 2004, pp. 369.
- <sup>7</sup>Messac, A., Ismail-Yahaya, A., Mattson, C.A., "The Normalized Normal Constraint Method for Generating the Pareto Frontier", *Structural and Multidisciplinary Optimization*, Vol. 25, 2003, pp. 86
- <sup>8</sup>Messac, A., Mattson, Christopher A., "Normal Constraint Method with Guarantee of Even Representation of Complete Pareto Frontier", *AIAA Journal*, Vol. 42, No. 10, 2004, pp 2101.
- <sup>9</sup>Du, X., Chen, Wei, "Collaborative Reliability Analysis under the Framework of Multidisciplinary Systems Design", *Optimization and Engineering*, Vol. 6, 2005, pp. 63
- <sup>10</sup>Koch, P. N., Yang, R. J., et al, "Design for six sigma through robust optimization", *Structural and Multidisciplinary Optimization*, Vol. 26, No. 3, 2004, pp. 235.
- <sup>11</sup>Wu, Y. T., *FPI User's and Theoretical Manuals*, Southwest Research Institute, San Antonio, Texas, 1995.
- <sup>12</sup>Wallace, J. M., "A Framework for Conducting Mechanistic Based Reliability Assessments of Components Operating in Complex Systems", Ph. D., School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, 2003.
- <sup>13</sup>Smaling, R. M., Weck, Olivier L. de, "Fuzzy Pareto Frontiers in Multidisciplinary System Architecture Analysis", 2004, AIAA-2004-4553.
- <sup>14</sup>Sues, R. H., Oakley, D. R., Rhodes, G. S., "MDO of Aeropropulsion Components Considering Uncertainty", 1996, AIAA-96-4062.
- <sup>15</sup>Dieter, G. E., *Engineering Design: A Materials and Processing Approach*, 3<sup>rd</sup> Edition, McGraw-Hill, 2000.
- <sup>16</sup>Marler, R. T., Arora, J. S., "Survey of Multi-Objective Optimization Methods for Engineering", *Structural and Multidisciplinary Optimization*, Vol. 26, 2004, pp. 369.
- <sup>17</sup>Yen, J., Wang, L., "Application of Statistical Information Criteria for Optimal Fuzzy Model Construction", *IEEE Transactions on Fuzzy Systems*, Vol. 6, No. 3, 1998, pp. 362.
- <sup>18</sup>Schwarz, G., "Estimating the Dimension of a Model", *The Annals of Statistics*, Vol. 6, No. 2, 1978, pp. 461.
- <sup>19</sup>Myers, R. H., Montgomery, D. C., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John Wiley & Sons, New York, 1995.
- <sup>20</sup>Smola, A. J., Scholkopf, B., et al, "The Connection Between Regularization Operators and Support Vector Kernels", *Neural Networks*, Vol. 11, No. 4, 1998, pp. 637
- <sup>21</sup>Shao, J., "Linear Model Selection via Cross-Validation", *Journal of the American Statistical Association*, Vol. 88, No. 422, 1993, pp. 486.
- <sup>22</sup>Lendasse, A., Simon, G., et al, "Fast Bootstrap Methodology for Regression Model Selection", *Neurocomputing*, Vol. 64, 2005, pp. 161.
- <sup>23</sup>Cherkassky, V., Ma, Yunqian, "Practical Selection of SVM Parameters and Noise Estimation for SVM Regression", *Neural Networks*, Vol. 17, 2004, pp. 113.
- <sup>24</sup>Shooman, M. L., *Probabilistic Reliability: an Engineering Approach*, 2nd edition, Krieger Publication Co., 1990.