N° d'ordre : 351

CENTRALE LILLE

THESE

Présentée en vue
d'obtenir le grade de

DOCTEUR

En

Spécialité : Génie Electrique

Par

DENG Siyang

DOCTORAT DELIVRE PAR CENTRALE LILLE

Titre de la thèse :

Méthodes de conception par optimisation robuste et fiable de dispositifs électromagnétiques

Methods for Robust and Reliability-based Design Optimization of Electromagnetic Devices

Soutenue le 22 janvier 2018 devant le jury d'examen :

Rapporteur	Arnaud HUBERT	Professeur, Université Technique de Compiègne
Président du jury et Rapporteur	Frédéric WURTZ	Directeur de Recherche, École nationale supérieure de l'énergie, l'eau et l'environnement
Examinateur	Carole HENAUX	Maître de Conférences HDR, ENSEEIHT
Examinateur	Jean-Christophe OLIVIER	Maître de Conférences HDR, Université de Nantes
Directeur de thèse	Stéphane BRISSET	Maître de Conférences HDR, Centrale Lille
Directeur de thèse	Stéphane CLENET	Professeur, École Nationale Supérieure d'Arts et Métiers

Thèse préparée dans le Laboratoire d'Electrotechnique et d'Electronique de Puissance de Lille
Ecole Doctorale Sciences pour l'ingénieur 072

Remerciement

Je tiens à exprimer ma sincère gratitude à tous ceux qui m'ont aidé ou encouragé au cours des trois dernières années.

Je souhaite en premier lieu remercier mes deux directeurs, Monsieur Stéphane Brisset et Monsieur Stéphane Clénet du Laboratoire d'électrotechnique et d'électronique de puissance de Lille (L2EP). Sans leurs aides, leurs encouragements, leurs conseils précieux et minutieux révisions, je n'aurais pas terminé mes études.

Je remercie Monsieur Arnaud Hubert, professeur à l'Université Technique de Compiègne et Monsieur Frédéric Wurtz, directeur de Recherche au Génie Électrique de Grenoble (G2eLab), pour avoir accepté d'être rapporteur de cette thèse et les causions scientifiques de mon travail.

Je tiens également à remercier Madame Carole Henaux, maître de conférences habilitée à diriger des recherches à l'Ecole Nationale Supérieure d'Electrotechnique, d'Eletronique, d'Informatique, d'Hydraulique et des Télécommunications (ENSEEIHT) et Monsieur Jean-Christophe Olivier, maître de conférences habilité à diriger des recherches à l'Institut de Recherche en Energie Electrique de Nantes Atlantique (IREENA), pour avoir participer à mon jury de thèse et pour l'intérêt qu'ils ont porté aux travaux que j'ai menés.

Je tiens ensuite à remercier tout le monde du L2EP, et plus particulièrement l'équipe Outils et Méthodes Numériques, qui m'a accueilli durant trois ans dans de parfaites conditions de travail.

Je voudrais également remercier Mojito, qui m'a soutenu dans les moments les plus difficiles et m'a aidé de lutter contre l'insomnie.

En cette de fin de thèse, je ne peux m'empêcher d'évoquer mes parents, ils ont toujours été là pour moi et qui m'ont toujours soutenue envers et contre tout, malgré des milles kilomètres qui nous séparent. Cette thèse leur est dédiée en premier.

Table of Contents

List of Tables

List of Figures

Acronyms

ANN	Artificial Neural Networks (a method to obtain a meta-model)
AM	Analytic Model
AMA	Approximate Moments Approach (a method of reliability-based design optimization using approximated constraints instead of probabilistic ones)
AMV	Advanced Mean Value (a method to calculate the maximum performance target point)
AWEI	Adaptive Weighted Expected Improvement
BLUP	Best Linear Unbiased Predictor
CDF	Cumulative Distribution Function
CEI	Constrained Expected Improvement
CMV	Conjugate Mean Value (a method to calculate the maximum performance target point)
DDO	Deterministic Design Optimization
EF	Expected Feasibility
EGO	Efficient Global Optimization (a method to solve optimization problems with meta-models)
EGRA	Efficient Global Reliability Analysis (a method to solve constrained optimization problems with meta-models)
EGRO-IE	Efficient Global Robust Optimization under Implementation Error
EHVI	Expected Hyper Volume Improvement
EI	Expected Improvement (a widely used infill searching criterion)
EV	Expected Violation (a type of infill searching criterion for constraints)
FEA	Finite Element Analysis
FEM	Finite Element Model
FORM	First-Order Reliability Method (a method uses to calculate an approximation of probability of failure)
GEI	Generalized Expected Improvement
GI	Gradient Index
GM	Global Minimum
GP	Gaussian Process
GWCO	Gradient-based Worst-Case Optimization
HL-RF	Hasofer Lind-Rackwitz Flessler (a method to calculate the reliability index)
HMV	Hybrid Mean Value (a method to calculate the maximum performance target point)
ISC	Infill Searching Criterion (criterion using in optimization with meta-models which can help to choose additional points)

LM	Local Minimum
LTB	Larger-The-Better (a formulation of reliability-based robust design optimization which intends to search the maximum of mean value while minimizing the standard deviation)
MARS	Multivariate Adaptive Regression Splines (a class of meta-model method)
MCS	Monte-Carlo Simulation (a simulation method which is used to calculate the probability with a huge quantity of samples)
MPP	Most Probable Point of failure (the point that minimizes the distance between the origin and the limit-state, used to calculate the reliability index for reliability index approach)
MPTP	Maximum Performance Target Point (the tangent point of targeted reliability surface and contours of constraint, used to calculate the maximal performance measurement for performance measurement approach)
MSE	Mean Square Error
MWEI	Modified Weighted Expected Improvement
NSGA	Non-dominated Sorting Genetic Algorithms
NTB	Nominal-The-Best (a formulation of reliability-based robust design optimization which aims to archive the target mean value while minimizing the standard deviation)
OK	Ordinary Kriging
PDF	Probability Density Function
PF	Probability of feasibility (a type of infill searching criterion for constraints)
PI	Probability of Improvement (a type of infill searching criterion)
PMA	Performance Measure Approach (a method of reliability-based design optimization using two loops)
RBDO	Reliability-Based Design Optimization (optimization which aims to reduce the probability of failure caused by uncertainties)
RBF	Radial Basis Function (a type of meta-model)
RBRDO	Reliability-Based Robust Design Optimization (optimization which considers both reliability and robust)
RDO	Robust Design Optimization (optimization which decides to find a solution insensible to the uncertainties)
RIA	Reliability Index Approach (a method of reliability-based design optimization using two loops)
RSM	Response Surface Methodology (a type of meta-model)
SAP	Sequential Approximate Programming (a sequential decoupled method of reliability-based design optimization)
SDO	Stochastic Design Optimization
SK	Simple Kriging

SLA	Single Loop Approach (a single loop method of reliability-based design optimization which uses approximated constraints instead of probabilistic ones)
SORA	Sequential Optimization and Reliability Assessment (a sequential decoupled method of reliability-based design optimization)
SQP	Sequential Quadratic Programming (an iterative method for constrained nonlinear optimization)
STB	Smaller-The-Better (a formulation of reliability-based robust design optimization which minimizes the mean value and standard deviation at the same time)
UK	Universal Kriging
WCEI	Worst-Case Expected Improvement
WCO	Worst-Case Optimization (optimization which makes sure even the worst-case caused by uncertainties can satisfy the uncertainties)
WEI	Weighted Expected Improvement
WWCO	Worst-vertex-based Worst-Case Optimization

Notations

a, b, c	Parameters for the shape of lamination of transformer
B_m	Maximal induction
d	Thickness of the frame of transformer
$\boldsymbol{d}$	Deterministic input variables
$\boldsymbol{d}^L$	Lower bound of deterministic input variables
$\boldsymbol{d}^U$	Upper bound of deterministic input variables
$\boldsymbol{d}_{w,\bullet}$	Design point correspondent to the worst-case
D_f	Failure domain
$E(\bullet)$	Expectation
E_i	Electromotive force
$f(\boldsymbol{d})$	Objective or cost function
$f_r(\boldsymbol{d})$	Robust objective function
$f_w(\boldsymbol{d})$	Worst-case of objective function
$\hat{f}(\boldsymbol{x})$	Predicted value of objective function at $\boldsymbol{x}$
f_i	Filling factors
f_{min}	Current best objective function value of all samples
f_t	Objective function target value
f_X	Probability density function
F_X	Cumulative distribution function
$g(\boldsymbol{d})$	Deterministic inequality constraints (should be negative)
$g_r(\boldsymbol{d})$	Robust inequality constraints
$g_w(\boldsymbol{d})$	Worst-case of inequality constraints
$\hat{g}(\boldsymbol{x})$	Predicted value of constraint at $\boldsymbol{x}$
g_{max}	Current maximum constraint value of all samples
G_p	Maximal performance measurement, given by PMA
$G_U(\boldsymbol{u})$	Performance function
$h(\boldsymbol{d})$	Deterministic equality constraints
I_1, I_2, I_μ	Primary, secondary and magnetizing currents
k	Confidence level (which is similar to the reliability index)
$K(\boldsymbol{x}, \boldsymbol{x}')$	Covariance function
l_i	Leakage inductance
l_{turn}	Average length of a coil turn
L_μ	Magnetizing inductance
L_2	Total magnetizing inductance
m	Number of constraints
m_{co}, m_{ir}	Copper and iron mass

m_{tot}	Total mass of transformer
$\boldsymbol{n}$	Normalized steepest ascent direction of constraint
n_i	Number of turns for coil i
N	Number of sample points
p	Dimension of input variables
P_{co}, P_{ir}	Copper and iron losses
P_f	Probability of failure
P_t	Target probability of failure
r_i	Resistor of coil i
R	Correlation function
R_{co}, R_{ir}, R_{air}	Thermal resistances of copper, iron and air
R_μ	Magnetizing resistor
R_2	Total magnetizing resistor
$\hat{s}(\boldsymbol{x})$	Mean square error of objective function at $\boldsymbol{x}$
$\hat{s}_g(\boldsymbol{x})$	Mean square error of constraint at $\boldsymbol{x}$
S	Search space
S_1, S_2	Primary and secondary section of conductors
$\boldsymbol{t}$	Shifting vector
T_{co}, T_{ir}, T_{am}	Copper, iron and ambient temperature
$\boldsymbol{u}$	Normalized input which follows the normal law
$\boldsymbol{u}^*$	Most Probable Point of failure
$U(\boldsymbol{d})$	Uncertainty set around the design point
V_i	Voltage
V_{20}	Voltage at no-load
$\boldsymbol{X}$	Input variables with uncertainties
$\boldsymbol{Z}$	Gaussian process
α	Weight parameter for infill searching criterion
α_{co}	Variation of the resistivity of copper
β	Reliability index (which is similar to the confidence level)
β_t	Target reliability index
ε	Small positive tolerance
Φ	Standard Gaussian cumulative distribution function
φ	Standard Gaussian probability density function
φ_1, φ_2	Total leakage fluxes of the primary and secondary coils
$\mu_f(\boldsymbol{d})$	Mean of objective function at $\boldsymbol{d}$
$\mu_f{}^w(\boldsymbol{d})$	Worst mean value of objective function among the uncertainty set $U(\boldsymbol{d})$
μ_{f_0}	Scaling value for mean of objective function
$\boldsymbol{\mu_g}(\boldsymbol{d})$	Mean of constraints $\boldsymbol{g}$ at $\boldsymbol{d}$
$\boldsymbol{\sigma}$	Standard deviation of uncertain input variables

$\sigma_f(\boldsymbol{d})$	Standard deviation of objective function at $\boldsymbol{d}$
σ_{f_0}	Scaling value for standard deviation of objective function
σ_{fmax}	Maximum of objective's standard deviation
σ_{fmin}	Minimum of objective's standard deviation
$\boldsymbol{\sigma_g}(\boldsymbol{d})$	Standard deviation of constraints at $\boldsymbol{d}$
ρ_{co}	Resistivity of copper
η	Efficiency
$\omega, \omega_1, \omega_2$	Weighted parameters for weighted robust formulation
$\partial g/\partial x_i$	First order derivative of constraint relative to input parameter x_i
$\nabla \bullet (\boldsymbol{d})$	Gradient function

Résumé étendu

Contexte

L'optimisation consiste à choisir un ensemble de variables qui optimisent les objectifs soumis aux contraintes de bon fonctionnement. Cependant, dans la vie réelle, les variables ne sont pas toujours déterministes car des incertitudes sont apportées par l'environnement, le modèle, la mesure et les dimensions. Dans cette thèse, on considère uniquement la dernière source d'incertitude. Ces dernières se propagent et peuvent provoquer des performances instables qui sont liées à la robustesse ou la violation des contraintes qui se rapportent à la fiabilité ainsi que d'autres conséquences inattendues. Il est donc important de prendre en compte les incertitudes pendant l'optimisation afin d'améliorer la faisabilité de la solution obtenue.

Aujourd'hui, de nombreuses universités et laboratoires travaillent sur l'optimisation dans le génie électrique, notamment pour la conception des dispositifs électromagnétiques.

Des chercheurs de l'université de l'Iowa en Les États-Unis travaillent sur des nouvelles méthodes de conception par optimisation fiable (reliability-based design optimization, RBDO) pour un processus en deux-niveaux. Ces méthodes sont utilisées pour la conception des dispositifs électromagnétiques dans des véhicules militaires à chenilles.

Les chercheurs de l'institut coréen de sciences et de technologie de l'information travaillent à l'innovation de SMES (Superconducting Magnetic Energy Storage System) Ils fournissent des services de modélisation et de simulation basés sur des super-ordinateurs qui raccourcissent considérablement la période de développement technologique et de réduire son coût. La méthode RIA est utilisée pour traiter les incertitudes.

L'université nationale de Chungbuk en Corée du sud propose une méthode qui combine RBDO et RDO (robust design optimization). Cet algorithme est proposé sur la base du pire scénario et fournit une approximation pour l'évaluation de la robustesse. La méthode SA-MCS (sensitivity-assisted Monte Carlo simulation) est aussi développée pour évaluer efficacement la fiabilité pour des problèmes électromagnétiques. L'algorithme est déjà appliqué au problème de TEAM 22 (SMES Optimisation Benchmark) et les résultats montrent qu'il peut trouver efficacement une bonne solution parmi toutes les caractéristiques de qualité.

L'université Tongji en Chine utilise l'optimisation probabiliste pour faire la conception avancée et la fabrication des carrosseries de véhicule. Les chercheurs proposent des méthodes RBDO ou RDO multi-objectif avec la méthode des surfaces de réponse en remplacent la simulation par la méthode des éléments finis pour améliorer l'efficacité. Ces

méthodes ont déjà été utilisées pour la conception d'une porte de véhicule et l'amélioration de la durée de vie d'une cabine de camion.

Il y a aussi d'autres pays qui travaillent sur ce domaine, surtout en France. Par exemple, à l'université de Grenoble, des chercheurs travaillent sur les moteurs électriques et les actionneurs électromagnétiques. A l'Université de Technologie de Compiègne, des chercheurs proposent des méthodes appliquées à un alterno-démarreur automobile et des concepteurs de l'Université de Toulouse optimisent la conception des avions.

Le but de cette thèse est de comparer les différentes méthodes et d'en faire une synthèse pour aider les concepteurs qui ne sont pas des spécialistes de l'optimisation à choisir les méthodes les plus appropriées à leurs besoins et les plus efficaces. Les modèles éléments finis étant très utilisés pour la conception des dispositifs électrotechniques, des nouvelles méthodes adaptées aux modèles lourds sont également proposées.

Méthodes d'optimisation avec incertitudes

Dans cette partie, les différentes catégories d'optimisation avec incertitudes sont présentées en détail puis l'approche la plus efficace dans chaque catégorie est mise en évidence.

Les méthodes d'optimisation avec incertitudes peuvent être divisées en quatre grandes catégories. L'optimisation dans le pire cas, robuste, fiable et puis robuste et fiable. L'optimisation dans le pire cas (worst-case optimization en anglais ou WCO en bref) utilise les bornes de distribution pour trouver le pire cas causé par l'incertitude. Optimisation robuste (robust design optimization en anglais ou RDO comme abréviation) se concentre sur la recherche d'une solution stable en utilisant la moyenne et l'écart-type des variables d'entrée. L'optimisation fiable (reliability-based design optimization en anglais ou RBDO) utilise la densité de probabilité des variables d'entrée pour calculer la probabilité de défaillance. Enfin, la dernière méthode RBRDO (abréviation de reliability-based robust design optimization) est une combinaison de robuste et fiable. Maintenant, on va présenter des détails de chaque catégorie.

Optimisation dans le pire cas (WCO)

L'optimisation dans le pire cas crée un ensemble d'incertitudes autour de chaque point de conception. Cet ensemble présente les performances possibles causées par les incertitudes. WCO s'assure que même les sorties les plus défavorables causées par des incertitudes peuvent respecter les contraintes.

On peut utiliser un exemple simple pour l'expliquer. Il comprend une variable d'entrée d et une contrainte g où la partie hachurée est la partie de défaillance, les segments autour des points de conception représentent les performances possibles causées par l'incertitude. Et on peut voir que l'optimum déterministe qui ne considère pas des

incertitudes ne satisfait pas la contrainte dans le pire cas car l'extrémité de segment c'est-à-dire le pire cas de ce point de conception est dans le domaine de défaillance. Par contre, le point bleu devient l'optimum dans le pire cas parce que le pire cas se situe dans l'état-limite $g = 0$ c'est-à-dire tous les situations possibles ne violent pas la contrainte.

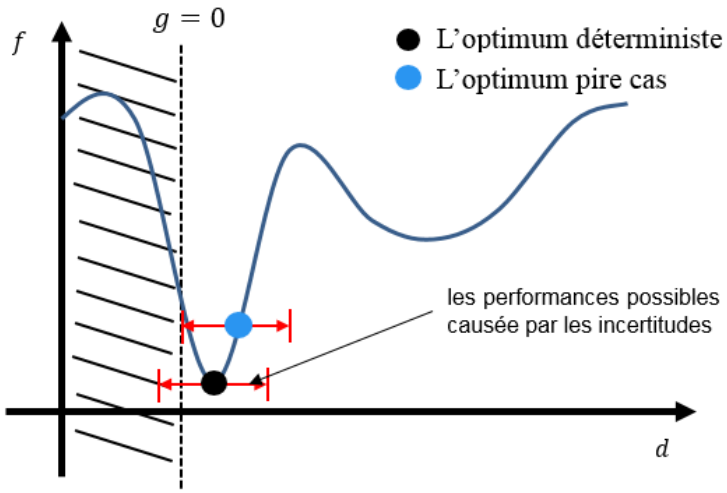


Fig. 1. L'optimum déterministe, l'optimum du pire cas et l'ensemble d'incertitude

Pour créer l'ensemble d'incertitude autour de point de conception, il y a plusieurs de définition. Dans cette thèse on utilise l'expression ci-dessous :

$$U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^p | \boldsymbol{d} - k\boldsymbol{\sigma} \leq \boldsymbol{\xi} \leq \boldsymbol{d} + k\boldsymbol{\sigma}\} \tag{1}$$

où k est le niveau de confiance décidé par le concepteur, k est plus grand signifie qu'il y a plus de possibilité que les valeurs d'entrée influencées par les incertitudes se situent dans cet ensemble.

La formulation d'optimisation dans le pire cas peut être exprimée comme un problème de minimax :

$$\begin{aligned} &\min_{\boldsymbol{d} \in S} f_w(\boldsymbol{d}) \\ &\text{soumis à } \boldsymbol{g}_w(\boldsymbol{d}) \leq 0 \\ &\text{avec } f_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} f(\boldsymbol{\xi}) \\ &\qquad \boldsymbol{g}_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \boldsymbol{g}(\boldsymbol{\xi}) \end{aligned} \tag{2}$$

Pour résoudre ce problème, cette thèse utilise trois différents types de méthodes. La première est WCO traditionnel avec deux boucles imbriquées. Elle utilise un sous problème d'optimisation pour trouver le pire des cas pour l'objectif et les contraintes. Cependant, comme il y a une boucle d'optimisation nichée dans une autre, ce processus nécessite un trop grand temps de calcul.

Afin de gagner du temps, le deuxième type de méthode n'évalue que les bornes. Il s'appelle Worst-vertex WCO en anglais ou WWCO. Le principe est d'évaluer les bornes de

chaque variable dans l'ensemble d'incertitude puis de déterminer les directions dans lesquelles les valeurs des fonctions augmentent. On suppose que le sommet qui se situe dans ces directions est le pire cas.

Le troisième type est Gradient-based WCO ou GWCO en bref. Il utilise le gradient et un développement de Taylor au premier ordre. Grâce à l'inégalité de Cauchy-Schwarz, la majoration du pire des cas peut être déduite et remplacer les pires performances réelles. Cette méthode utilise encore moins d'évaluations que WWCO mais en raison de la majoration, les pires valeurs dépassent quelques fois l'ensemble d'incertitude.

Donc, en considérant la précision et la vitesse de convergence, WWCO est le plus efficace entre ces trois types.

Optimisation robuste (RDO)

La deuxième catégorie d'optimisation avec incertitudes est l'optimisation robuste. Robustesse signifie que la solution est moins sensible aux incertitudes sur les entrées, c'est-à-dire qu'un petit changement d'entrée n'apporte pas de grandes variabilités de sortie. On peut utiliser le même exemple unidimensionnel pour l'expliquer :

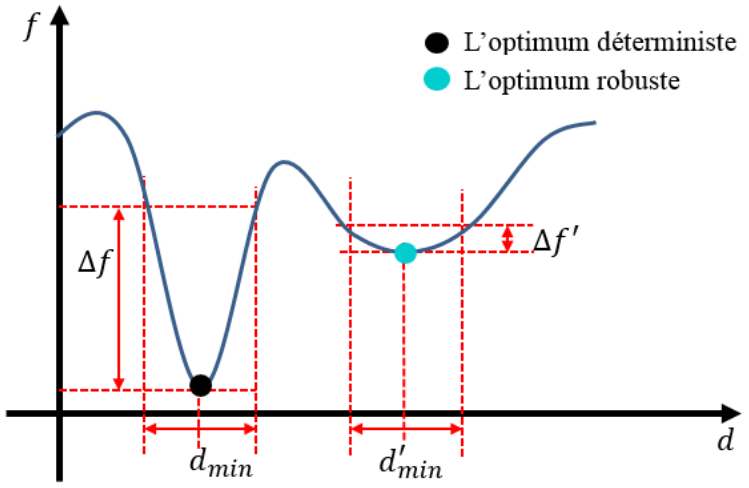


Fig. 2. L'optimum déterministe et l'optimum robuste

d_{min} et d'_{min} sont l'optimum global et un optimum local respectivement, Δf et $\Delta f'$ sont les variances de sortie causées par les incertitudes d'entrée autour de d_{min} et d'_{min}. Δf est beaucoup plus grande que $\Delta f'$ donc d'_{min} est moins sensible ou plus robuste que d_{min}.

Ainsi le but de l'optimisation robuste est de minimiser la fonction objectif et sa variance en même temps. C'est donc un problème multi-objectif et la formulation est écrite comme suit :

$$\begin{aligned} &\min_{\boldsymbol{d}} f_r\left(\mu_f(\boldsymbol{d}), \sigma_f(\boldsymbol{d})\right) \\ &\text{soumis à} \quad \boldsymbol{g_r}\left(\boldsymbol{\mu_g}(\boldsymbol{d}), \boldsymbol{\sigma_g}(\boldsymbol{d})\right) \leq 0 \\ &\boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{aligned} \tag{3}$$

où $\boldsymbol{d}^L$ et $\boldsymbol{d}^U$ sont des bornes inférieure et supérieure des variables d'entrée, k est le niveau de confiance comme dans l'optimisation dans le pire cas, f_r et $\boldsymbol{g_r}$ sont la fonction objectif robuste et les contraintes robustes calculées à partir des moments statistiques $\mu_f, \sigma_f, \mu_g, \sigma_g$.

Pour résoudre ce problème, 9 différentes formulations sont utilisées et comparées dans cette thèse. Elles peuvent être divisées en deux sous-types de méthodes : le premier type est mono-objectif et transforme la formulation en adjoignant le deuxième objectif au premier. Les trois transformations testées dans la thèse sont présentées dans le tableau suivant :

Tableau 0. Formulations pour des méthodes mono-objectif

Formulation 1	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + k\boldsymbol{\sigma_g}(\boldsymbol{d})$
Formulation 2	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d}) + k\sigma_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + k\boldsymbol{\sigma_g}(\boldsymbol{d})$
Formulation 3	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d})/\lvert\mu_{f_0}\rvert + k\,\sigma_f(\boldsymbol{d})/\sigma_{f_0}$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + k\boldsymbol{\sigma_g}(\boldsymbol{d})$

Avec ce tableau on peut voir que ces transformations fixent le poids entre la moyenne et l'écart-type. Donc, elles ne trouvent qu'une seule solution.

Puis un autre type de méthodes est multi-objectif et traite la moyenne et l'écart-type comme deux objectifs différents afin d'obtenir un front de Pareto. Six formulations appartenant à 3 différentes méthodes sont présentées. La première est ε-contrainte et transforme le problème multi-objectif en une série de problèmes mono-objectif avec une contrainte supplémentaire de sorte qu'un algorithme mono-objectif est adapté. La deuxième est la méthode pondérée qui combine aussi deux objectifs ensemble mais cette fois avec un poids qui change sa valeur entre 0 à 1 pour obtenir différentes solutions. La dernière approche ne change pas la formulation mais elle utilise des algorithmes multi-objectif directement. Les formulations principales sont présentées dans le tableau suivant :

Tableau 2. Formulations pour des méthodes multi-objectif

ε-contrainte	$\boldsymbol{f_r}(\boldsymbol{d}) = \boldsymbol{\mu_f}(\boldsymbol{d})$	$\boldsymbol{g_{r_1}}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \boldsymbol{k\sigma_g}(\boldsymbol{d})$ $\boldsymbol{g_{r_2}}(\boldsymbol{d}) = \boldsymbol{\sigma_f}(\boldsymbol{d}) - \boldsymbol{\sigma_t}$, $\boldsymbol{\sigma_t} \in [\boldsymbol{\sigma_{fmin}}, \boldsymbol{\sigma_{fmax}}]$
méthode pondérée	$\boldsymbol{f_r}(\boldsymbol{d}) = \boldsymbol{\omega\mu_f}(\boldsymbol{d}) + (\boldsymbol{1} - \boldsymbol{\omega})\boldsymbol{\sigma_f}(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \boldsymbol{k\sigma_g}(\boldsymbol{d})$
algorithmes multi-objectif	$\boldsymbol{f_{r1}}(\boldsymbol{d}) = \boldsymbol{\mu_f}(\boldsymbol{d}), \boldsymbol{f_{r2}}(\boldsymbol{d}) = \boldsymbol{\sigma_f}(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \boldsymbol{k\sigma_g}(\boldsymbol{d})$

$\boldsymbol{\sigma_{fmin}}, \boldsymbol{\sigma_{fmax}}$ sont les bornes de $\boldsymbol{\sigma_f}$, $\boldsymbol{\omega}$ est le poids pour la pondération. En ajoutant la normalisation, on peut avoir 6 formulations différentes. Entre ces trois approches, ε-contrainte peut toujours trouver un front par contre le résultat de la méthode de somme pondérée ne fonctionne qu'avec un front convexe et l'algorithme de multi-objectifs est plus lent, donc le premier est le plus efficace.

Optimisation fiable (RBDO)

La troisième catégorie d'optimisation avec incertitude est l'optimisation fiable. Elle fixe une valeur probabiliste maximale acceptable pour remplacer la contrainte initiale et la probabilité de défaillance doit être plus petite que cette valeur donnée. La probabilité de défaillance est la probabilité que la valeur de contrainte est supérieure à 0.

Dans la figure ci-dessous, le point rouge est un point de conception. On dessine la densité de probabilité sur ce point et la probabilité de défaillance est l'intégration de la densité de probabilité sur la partie rouge.

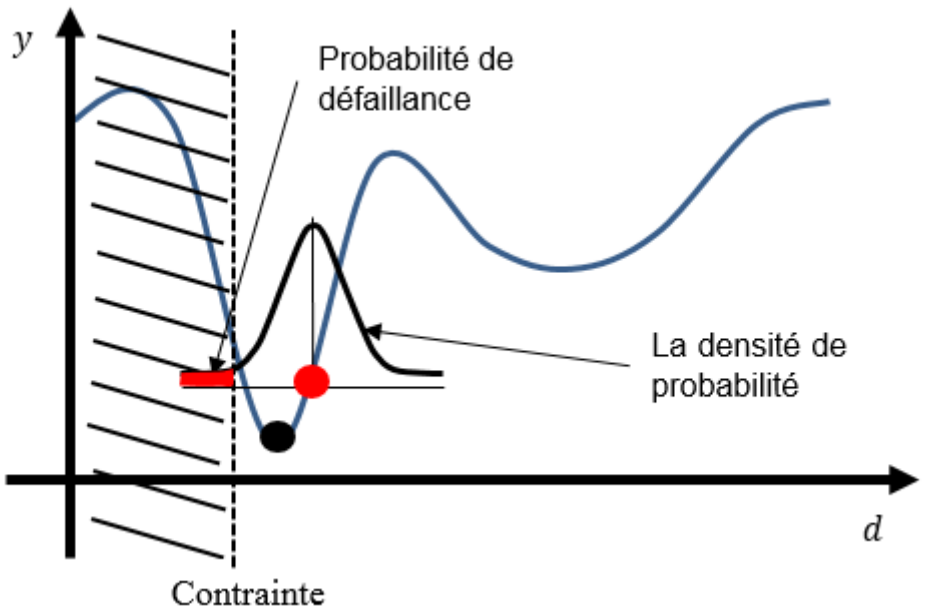


Fig. 3. La probabilité de défaillance

Le but de l'optimisation fiable est de trouver une solution dont les probabilités de défaillance pour toutes les contraintes respectent les nouvelles contraintes probabilistes. La formulation est changée par l'expression ci-dessous :

$$\begin{gathered} \min_{d} f(d) \\ \text{soumis à} \quad P_f = P(g(X) > 0) \leq P_t \\ d^L + k\sigma \leq d \leq d^U - k\sigma \end{gathered} \tag{4}$$

où P_f est la probabilité de défaillance et P_t est la probabilité cible fixée.

Selon les différentes façons de calculer les probabilités, RBDO peut être divisé en trois différents types de méthodes. Le premier s'appelle les méthodes de double-boucle et utilise la méthode de fiabilité du premier ordre pour calculer la probabilité de défaillance dans la boucle intérieure. La boucle extérieure a pour objectif de chercher les optima en utilisant la probabilité calculée par la boucle intérieure. Les deux approches les plus utilisées sont présentées dans cette thèse. RIA (reliability index approach) cherche itérativement le point de conception avec la plus forte probabilité de défaillance dans la boucle intérieure. L'objectif de la boucle intérieure de PMA (Performance Measure Approach) est de trouver le point de performance (contrainte) maximale correspondant à l'indice de fiabilité cible. PMA est plus robuste et plus efficace quand la contrainte probabiliste est inactive alors que RIA est plus efficace pour une contrainte probabiliste violée mais elle pourrait donner une singularité lorsque la contrainte est inactive.

Le deuxième type de méthodes utilise une approximation pour calculer la probabilité de défaillance. Ainsi, la boucle intérieure n'est plus une optimisation et c'est la raison pour laquelle ces méthodes s'appellent les méthodes simple boucle. Deux approches sont présentées en détail dans cette thèse : AMA (Approximate Moments Approach) et SLA (Single Loop Approach). La différence entre ces deux méthodes est que SLA évalue les contraintes à l'approximation du point de performance maximale contrairement à AMA qui utilise une approximation de premier ordre de la contrainte autour de la valeur moyenne.

Enfin, le troisième type de méthodes est séquentiel découplé et vise à changer le problème initial en une série de cycles d'optimisation. Chaque cycle séquentiel comprend une optimisation déterministe et une analyse de fiabilité. Deux méthodes SORA (Sequential Optimization and Reliability Assessment) et SAP (Sequential Approximate Programming) sont présentées. La seconde méthode utilise des approximations et elle est donc plus rapide que la première mais elle peut ne pas converger si le problème est complexe.

Après la comparaison de ces approches, les résultats montrent que la méthode séquentielle est la plus efficace notamment SORA alors que les méthodes double boucle ont besoin de trop de temps et les méthodes simple boucle ne sont pas assez précises.

Optimisation robuste et fiable (RBRDO)

La dernière catégorie RBRDO est un mélange de deux catégories précédentes. Le but de la robustesse est de trouver une solution dont la variabilité de l'objectif est faible alors que la fiabilité trouve une solution éloignée de l'état limite des contraintes. Il est donc intéressant d'avoir une formulation qui peut conduire à un optimum robuste et fiable à la fois.

Comme pour les méthodes RDO, les méthodes RBRDO exploitent des formulations mono-objectif et multi-objectif. Les premières ne trouvent qu'une seule solution. Pour les

secondes, il y a trois formulations selon les objectifs de conception. Nominal-the-best trouve la solution dont la valeur de l'objectif est la plus proche d'une valeur cible tout en minimisant l'écart-type de l'objectif. Smaller-the-better minimise la moyenne et l'écart-type de l'objectif en même temps. Enfin, larger-the-better vise à maximiser la moyenne et minimiser l'écart-type. Comme dans notre cas on ne considère que des problèmes de minimisation, seules les deux premières formulations sont adaptées. Les formulations sont présentées dans le tableau ci-dessous :

Tableau 3. Formulations pour des méthodes RBRDO

Nominal-the-best	$f_r(\mu_f, \sigma_f) = \omega_1 \left(\frac{\mu_f - f_t}{\mu_{f_0} - f_t}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$\boldsymbol{g_r(d) = \mu_g(d) + k\sigma_g(d)}$
Smaller-the-better	$f_r(\mu_f, \sigma_f) = \omega_1 \cdot sign(\mu_f) \left(\frac{\mu_f}{\mu_{f_0}}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$\boldsymbol{g_r(d) = \mu_g(d) + k\sigma_g(d)}$
Larger-the-better	$f_r(\mu_f, \sigma_f) = \omega_1 \cdot sign(\mu_f) \left(\frac{\mu_{f_0}}{\mu_f}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$\boldsymbol{g_r(d) = \mu_g(d) + k\sigma_g(d)}$

Synthèse

Après la présentation des quatre catégories d'optimisation intégrant les incertitudes, une synthèse est faite pour aider les concepteurs à choisir la méthode la plus appropriée à leurs besoins. WCO trouve une solution au problème de conception qui minimise l'objectif tout en respectant les contraintes dans le pire des cas. C'est une optimisation qu'on peut qualifier de conservatrice et peut conduire à un surdimensionnement. RDO est capable de trouver une conception peu sensible aux incertitudes et donc une performance plus stable. L'objectif principal de RBDO est d'identifier un optimum satisfait une probabilité de défaillance donnée. Il convient à une conception en grande série à moindre coût avec un taux de défaillance faible. RBRDO se concentre sur l'obtention d'une conception robuste et fiable en même temps.

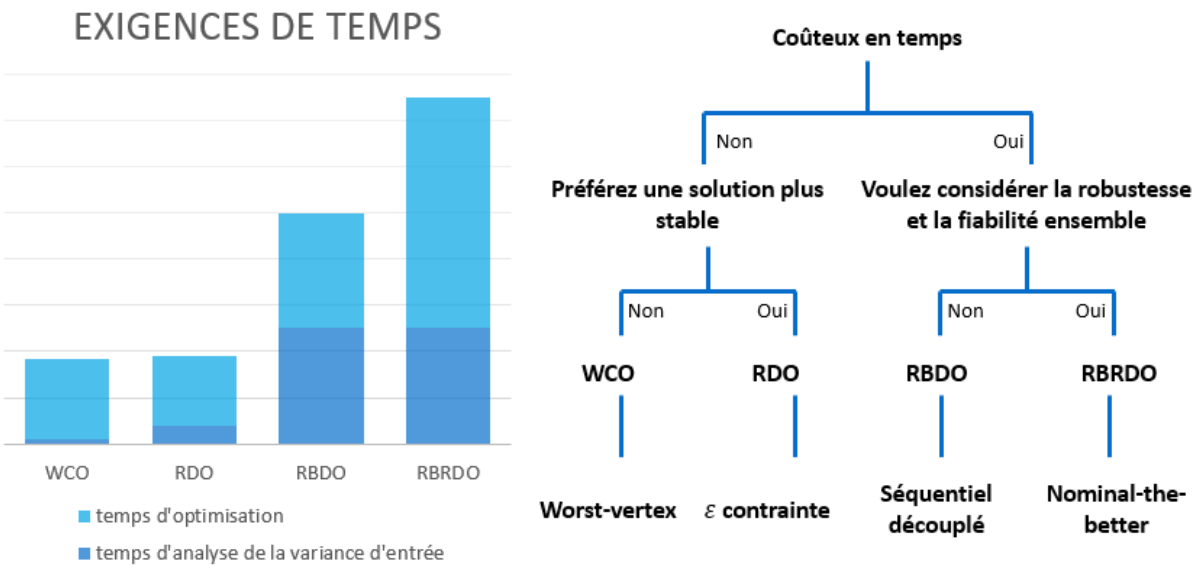


Fig. 4. Les exigences de temps pour chaque catégorie d'optimisation avec incertitudes

Les objectifs de conception variant selon les catégories type de méthodes, il n'est pas pertinent de prouver qu'une catégorie domine les autres. Par contre, on peut les classer par rapport aux temps nécessaires à la quantification des incertitudes et à l'optimisation.

La figure ci-dessus montre les exigences de temps pour les quatre catégories et le guide pour aider les concepteurs à choisir la méthode appropriée. La figure indique le temps d'optimisation et aussi le temps d'analyse de la variance d'entrée, c'est-à-dire l'analyse des extrema, des moments statistiques ou des densités de probabilité qui sont nécessaires aux optimisations spécifiques. Puisque WCO ne demande que des extrema, il n'a besoin que de très peu de temps pour l'analyse alors que pour RBDO et RBRDO, le temps d'analyse augmente pour obtenir des histogrammes approchant les densités de probabilité. Avec cette figure, les concepteurs peuvent choisir la catégorie d'optimisation avec incertitudes pour traiter leurs problèmes de conception selon le temps dont ils disposent. Si l'échantillonnage de variables d'entrée du système prend du temps, c'est mieux de choisir les méthodes qui n'ont pas besoin d'un trop grand échantillon pour l'analyser les variances. Dans ce cas, si les concepteurs veulent avoir une solution moins sensible, RDO est la méthode la plus appropriée, sinon ils peuvent se contenter de WCO qui sera encore plus rapide. Par contre, si le temps n'est pas un problème, les deux autres méthodes peuvent être utilisées : RBDO est plus appropriée si l'on se concentre sur la probabilité de défaillance, sinon RBRDO est plus général.

Optimisation avec méta-modèle

Les méthodes précédentes ne sont adaptées que pour des modèles rapides. Si le modèle est lourd comme dans le cas des éléments finis, l'optimisation prendra beaucoup trop de temps. Ainsi, on a besoin de trouver de nouvelles méthodes capables de traiter des modèles numériques lourds.

Le méta modèle est choisi pour nous aider. L'objectif du méta-modèle est de quantifier la structure spatiale d'une fonction et de prédire une valeur avec des échantillons limités. Il existe plusieurs méthodes de méta-modèle comme, par exemple, la méthode des surfaces de réponses, les fonctions de base radiale et le krigeage. Comme cette dernière est une méthode très utilisée, elle est choisie dans cette thèse. Le modèle est composé de deux parties : la moyenne $\mu_Y(\boldsymbol{x})$ est la partie déterministe et le processus gaussien $Z(\boldsymbol{x})$ est la partie stochastique. Ainsi Y $(\boldsymbol{x})$ est la prédiction à la position $\boldsymbol{x}$:

$$Y(\boldsymbol{x}) = \mu_Y(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{5}$$

Pour combiner le méta-modèle et l'optimisation, il faut créer une stratégie qui peut trouver l'optimum global et améliorer la précision de méta modèle en même temps. Nous avons donc besoin d'un critère d'enrichissement pour nous indiquer où ajouter de nouveaux échantillons.

Critères d'enrichissement

Le critère le plus célèbre s'appelle expected improvement ou EI en bref :

$$EI(\boldsymbol{d}) = \left[f_{min} - \hat{f}(\boldsymbol{d})\right]\Phi\left(\frac{f_{min}-\hat{f}(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) \quad + \quad \hat{s}(\boldsymbol{x})\phi\left(\frac{f_{min}-\hat{f}(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) \tag{6}$$

où Φ et ϕ représentent la fonction de répartition et la densité de probabilité pour la distribution normale centrée réduite et f_{min} est la meilleur valeur courante, $\hat{f}(\boldsymbol{d})$ est la valeur prévu, $\hat{s}(\boldsymbol{d})$ représente l'erreur.

La première partie d'EI présente la probabilité que la valeur de $\hat{f}$ est plus petite que f_{min}, elle favorise l'exploitation. La deuxième partie favorise l'exploration car elle augmente si l'erreur est plus grande.

En maximisant le critère, on a la capacité de trouver le point qui a la plus grande probabilité d'améliorer la précision et de trouver le minimum simultanément. Le maximum est ajouté dans l'ensemble d'échantillon et puis le méta-modèle est reconstruit. Cette boucle continuera jusqu'à ce qu'un critère d'arrêt soit satisfait.

EI est adapté à un problème déterministe sans contrainte, ainsi certains changements doivent être faits pour l'adapter à l'optimisation avec incertitude. Commençons par WCO qui consiste à résoudre un problème minimax : La méthode proposée est de trouver le pire cas, c'est à dire la plus petite valeur, d'EI dans l'ensemble d'incertitude U autour de d :

$$WCEI(\boldsymbol{d}) = \min_{\boldsymbol{\xi} \in U(\boldsymbol{d})} [EI(\boldsymbol{d} + \boldsymbol{\xi})] \tag{7}$$

où $U(\boldsymbol{d})$ est l'ensemble d'incertitude autour de $\boldsymbol{d}$.

Pour les autres catégories d'optimisation avec incertitudes, un poids α est ajouté afin de répartir l'attention d'EI sur l'exploitation et l'exploration.

$$WEI(\boldsymbol{d}) = \alpha\left[f_{min} - \hat{f}(\boldsymbol{d})\right]\Phi\left(\frac{f_{min}-\hat{f}(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) + (1-\alpha)\hat{s}(\boldsymbol{d})\phi\left(\frac{f_{min}-\hat{f}(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) \tag{8}$$

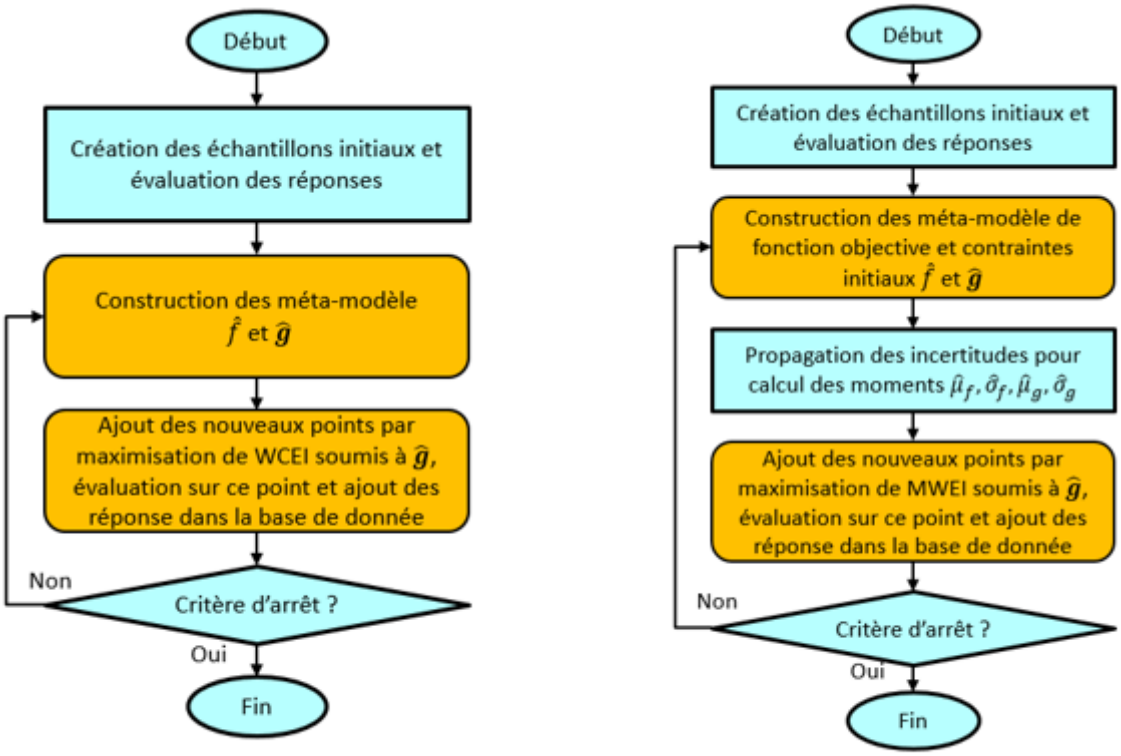
Il peut arriver que le domaine de sécurité soit trop petit et qu'aucun point dans l'échantillon initial ne respecte les contraintes. Dans ce cas, le critère ne marche pas car f_{min} est indéterminé ou trop petit. Pour éviter ce problème, un nouveau critère est proposé dans cette thèse : Modified Weighted EI (MWEI) ajoute un terme à (8) pour que sa valeur ne soit jamais nulle ou indéterminée.

$$MWEI = WEI(\boldsymbol{d}) - \alpha\hat{f} \tag{9}$$

Comme il y a des contraintes dans le problème et qu'elles sont longues à calculer, l'utilisation des contraintes réelles prend trop de temps. En cohérence avec les recommandations dans la littérature et après quelques essais, les méta-modèles des contraintes sont utilisés directement.

Optimisation dans le pire cas

Différentes stratégies sont proposées pour chaque catégorie d'optimisation avec incertitudes pour les adapter aux modèles lourds. Pour WCO, après une comparaison, le processus le plus efficace est présenté dans la Fig. 5a. Après le tirage d'un échantillon initial et la construction des méta-modèles, les nouveaux points sont ajoutés par maximisation du critère WCEI. Puis les méta-modèles sont reconstruits jusqu'à satisfaction d'un critère d'arrêt.



(a) le processus de WCO (b) le processus de RDO

Fig. 5. Les processus de WCO et RDO avec méta-modèle

Optimisation robuste

Pour la stratégie de RDO, on a besoin des moyennes et écart-type des objectifs et contraintes. Ainsi, il faut ajouter une autre couche de transformation pour laisser le critère s'adapter à la formulation.

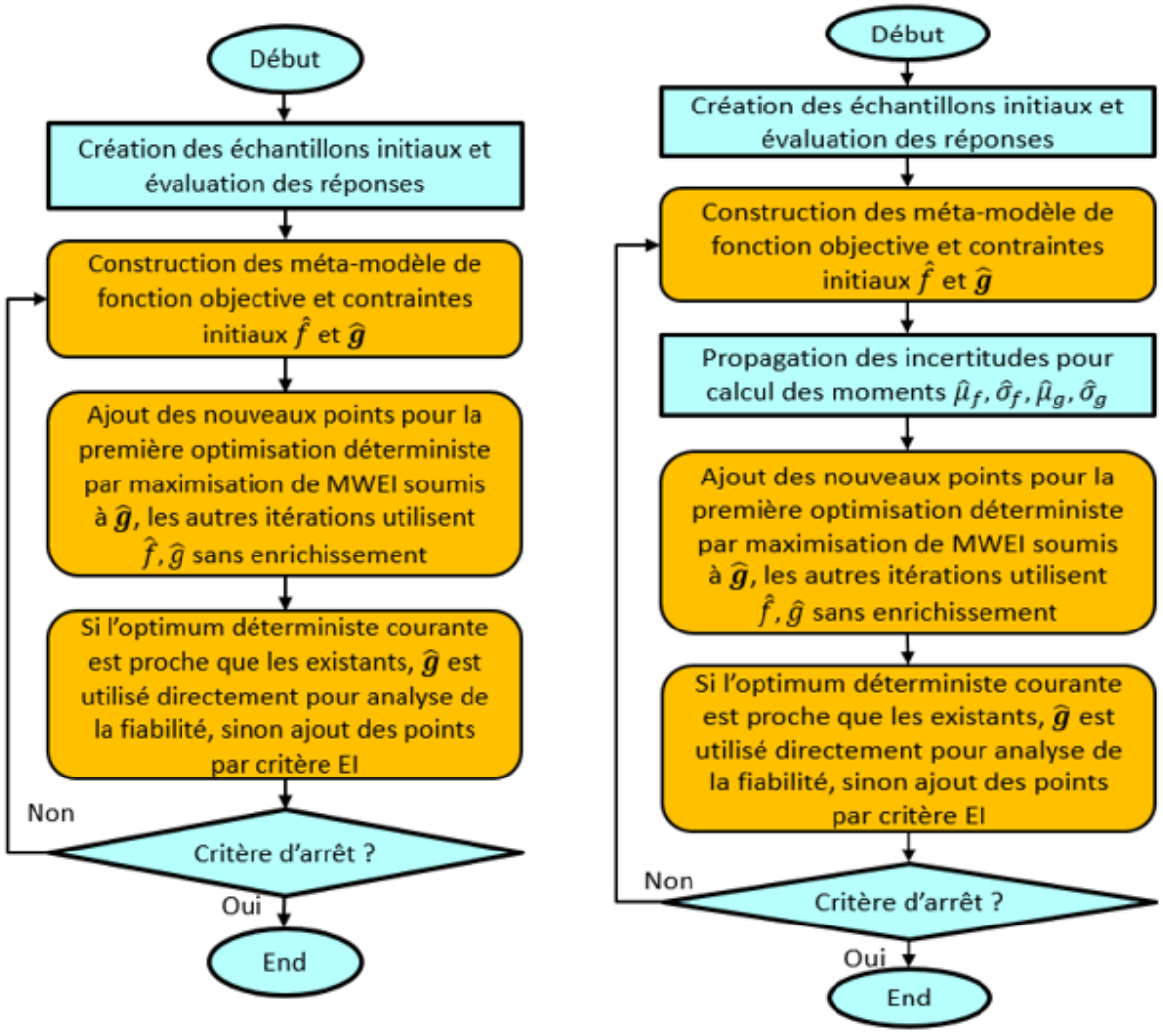
Une fois les méta-modèles $\hat{f}$ et $\hat{g}$ construits, on les utilise pour calculer les moments des fonctions objectif et contraintes. Ensuite, l'objectif et les contraintes robustes sont calculés avec ces moments et le critère MWEI est calculé. Le processus est présenté dans la Fig. 5b.

Optimisation fiable

Après avoir adapté les trois types de méthodes aux méta-modèles, une comparaison a montré que la méthode séquentielle SORA est la plus efficace. Son processus est présenté dans la Fig. 6a. Comme elle sépare l'optimisation déterministe et l'analyse de fiabilité, il y a deux places possibles pour l'enrichissement.

MWEI est utilisé pour la première itération d'optimisation déterministe car son but est de trouver l'optimum déterministe global. Pour les autres itérations, les méta-modèles $\hat{f}$ et $\hat{\boldsymbol{g}}$ sont utilisés directement sans enrichissement.

Pour l'analyse de fiabilité, on ajoute un critère de proximité : Si l'optimum trouvé dans l'itération courante est loin des autres évaluations du modèle lourd, on utilise critère EI pour faire un enrichissement, sinon on suppose que le méta-modèle est suffisamment précis pour être utilisé directement.



(a) le processus de RBDO

(b) le processus de RBRDO

Fig. 6. Les processus de RBDO et RBRDO avec méta-modèle

Optimisation robuste et fiable

Puisque RBRDO est une combinaison de RBDO et RDO, son processus est aussi un mélange de ces deux précédents. Son processus est présenté dans la Fig. 6b.

La fonction objectif robuste et les contraintes robustes sont calculées par les moments après que les méta-modèles $\hat{f}$ et $\hat{\boldsymbol{g}}$ soient construits. Ensuite, le critère MWEI est utilisé pour l'optimisation déterministe lors de la première itération et EI est utilisé pour analyse de la fiabilité si le critère de proximité est inactif.

Synthèse

Un exemple simple est utilisé pour tester la performance des méthodes proposées. Les résultats montrent que le nombre d'évaluations est grandement diminué mais que l'utilisation d'un méta-modèle entraine une perte de précision.

Un autre avantage des méta-modèles est de fournir des fonctions non-bruitées et leurs dérivées qui peuvent être utilisées avec des méthodes d'optimisation à base de gradient.

Application en électromagnétique

Afin de vérifier l'adéquation des méthodes proposées à la conception par optimisation des dispositifs électromagnétiques, le cas-test du transformateur monophasé introduit par Tuan-Vu TRAN est traité.

Il possède certaines propriétés qui rendent son optimisation difficile : Il est multiphysique, les équations sont multimodales et implicites, il y a beaucoup de contraintes actives à l'optimum, et les variables ont des ordres de grandeur forts différents.

Les phénomènes thermiques, magnétiques et électriques du transformateur sont modélisés par des modèles analytique et éléments finis.

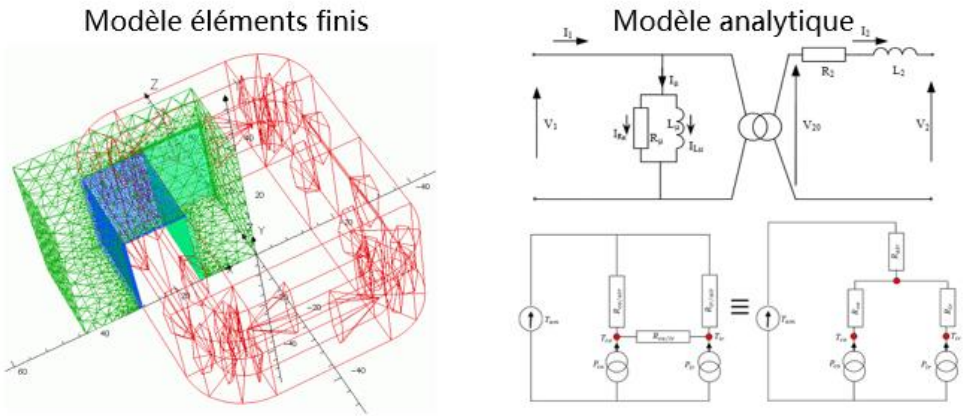


Fig. 7. Les modèles éléments finis et analytique du transformateur

Le problème d'optimisation déterministe est de minimiser la masse totale des parties actives (cuivre et fer). La formulation est présentée comme :

$$\begin{aligned} &min\, m_{tot}(a,b,c,d,n_1,S_1,S_2) \\ &s.\text{à}\ \ \boldsymbol{g}(a,b,c,d,n_1,S_1,S_2) \leq 0 \end{aligned} \tag{10}$$

$a, b, c, d, n_1, S_1, S_2$ sont les variables de conception, les quatre premières sont associées au circuit magnétique et les trois dernières se rapportent aux bobines primaire et secondaire. Les contraintes sont :

$$\boldsymbol{g} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \\ g_6 \\ g_7 \end{bmatrix} = \begin{bmatrix} T_{co} - 120°\text{C} \\ T_{ir} - 100°\text{C} \\ \frac{I_\mu}{I_1} - 0.1 \\ \frac{\Delta V_2}{V_2} - 0.1 \\ f_1 - 0.5 \\ f_2 - 0.5 \\ 0.8 - \eta \end{bmatrix} \tag{11}$$

Pour vérifier l'adéquation et l'efficacité des méthodes proposées, on suppose que chaque variable de conception suit la loi normale avec une valeur moyenne à trouver et un écart-type valant 1% de la borne inférieure. La probabilité de défaillance cible est fixée à 0.13%.

Modèle analytique

Pour le modèle analytique, un modèle thermique nodal et un modèle magnéto-électrique sont considérés. Puisque ce modèle est rapide et non-bruité, on utilise des méthodes sans méta-modèle.

Parmi les méthodes qui avaient donné des bons résultats sur l'exemple simple, la moitié d'entre-elles sont en défaut sur le cas-test du transformateur. Les distances relatives entre ces solutions sont montrées dans le Tableau 4.

Tableau 4. Distances relatives des solutions pour le modèle analytique

Méthode	Déterministe	WCO	RDO	RBDO
RBRDO	3.51%	2.83%	0.10%	0.08%
RBDO	3.57%	2.76%	0.17%	--
RDO	3.41%	2.92%	--	--
WCO	6.32%	--	--	--

Les valeurs sont calculées par l'expression ci-dessous qui présente le ratio de la distance entre les solutions sur la longueur de la diagonale de l'espace de recherche :

$$distance = \sqrt{\sum \left(\frac{d_1(i) - d_2(i)}{d^U(i) - d^L(i)} \right)^2} \quad (12)$$

On peut voir que les optima de RDO, RBDO et RBRDO sont similaires et distants de ceux de WCO et du problème déterministe. La distance plus petite est entre RBDO et RBRDO et la plus grande est entre WCO et déterministe. Ces résultats soulignent l'importance de considérer les incertitudes lors de la conception par optimisation.

Modèle éléments finis

Le modèle éléments finis comprend un 8ième du transformateur en raison des symétries. Le maillage contient environ quarante-trois mille nœuds et deux cent quatre-vingt mille éléments. Puisque le temps de calcul du modèle éléments finis est supérieur à dix minutes, on utilise des méta-modèles pour mener des optimisations en un temps raisonnable.

Un échantillon initial de 7000 points est utilisé pour construite les méta-modèles de l'objectif et des contraintes. Cette taille importante s'explique par le nombre de variables, les fonctions multimodales, et le problème fortement contraint qui induit un espace de faisabilité petit et éventuellement discontinu.

Les valeurs optimales trouvées par les différentes méthodes sont toujours distinctes de celles du modèle analytique car ce dernier sous-estime les contraintes. Les méthodes avec méta-modèle réduisent le nombre d'évaluations. Les distances relatives entre les solutions avec le modèle éléments finis sont montrées ci-dessous :

Tableau 5. Distances relatives des solutions avec le modèle éléments finis

Méthode	Déterministe	WCO	RDO	RBDO
RBRDO	4.83%	8.29%	4.29%	4.68%
RBDO	7.36%	11.81%	7.45%	--
RDO	1.41%	4.98%	--	--
WCO	5.33%	--	--	--

On peut voir que les distances sont différentes de celles avec le modèle analytique. Les optima avec incertitude sont éloignés les uns des autres. La distance la plus grande est entre RBDO et WCO. Cette fois, les résultats de RDO et déterministe sont proches. C'est en partie une conséquence de l'imprécision du méta-modèle.

Conclusion

Dans cette thèse, une comparaison des différentes catégories d'optimisation avec incertitudes est faite pour mettre en évidence les plus efficaces. Des nouveaux critères d'enrichissement du méta-modèle sont proposés pour améliorer la précision et réduire le nombre d'évaluations. L'utilisation conjointe des méthodes d'optimisation avec incertitude et des méta-modèles permet de gagner du temps lors de l'optimisation avec des modèles lourds. Enfin, ces méthodes sont appliquées avec succès dans domaine électromagnétique pour l'optimisation d'un transformateur monophasé.

Mais il y a encore beaucoup de points à améliorer. Premièrement, pour les critères d'enrichissement, on ajoute un seul échantillon après chaque itération, ce qui n'exploite pas le calcul parallèle. Si on ajoute plusieurs points à chaque itération, la précision du méta-modèle sera plus vite améliorée. Deuxièmement, pour le modèle éléments finis du transformateur, l'échantillon initial représente plus de 90% des évaluations totales alors que l'enrichissement représente moins de 10%. Cette répartition pourrait être plus équilibrée par un échantillon initial plus approprié. Troisièmement, les incertitudes sur les variables ont toujours été considérées indépendantes et suivant une distribution normale. Cependant, dans la vie réelle, ces hypothèses sont rarement vérifiées et il faudrait faire les changements appropriés dans les méthodes afin de pouvoir les appliquer plus largement.

General Introduction

Over the past few decades, as the high-speed digital computers made implementation of the optimization and simulation procedures possible, classical optimization methods has been applied successfully in all engineering fields, including electromagnetic engineering. Designs of complex electromagnetic devices can be modelled and the optima which can reduce the cost or improve the performance most can be found through simulation.

However, most of these methods are deterministic as they try to find the global optimum under the hypothesis that all input parameters are deterministic and no uncertainty is involved into the manufacturing process. The influence of this hypothesis may be insignificant for simple products or those having great tolerance, but for electromagnetic devices which are complicated and require a high accuracy, this assumption may cause undesirable consequences. In real conditions, many variables of the design problems are random and uncontrollable, the unexpected uncertainties will lead to a bias of performances, and in such cases, the optimum obtained by a deterministic design optimization (DDO) may violate the constraints unacceptably. If this optimum is kept, it may result in performance degradation or even worse in some unsafe situation.

Moreover, as all optimization methods rely on repeating evaluations of the models (Moustapha, Sudret, Bourinet, & Guillaume, 2016), the computational burden becomes insufferable if the model is too heavy to evaluate. So that the applications of deterministic models and deterministic optimizations still have great limitations in electromagnetic engineering.

In order to avoid the unexpected consequences resulted from uncertainties and to deal with complicate devices or systems, two main difficulties need to be faced. The first one is to get an optimum in design optimization that can always respect the constraints even if the uncertainties are taken into account. The second one is to reduce the computational cost as much as possible for high precision models especially when finite element analysis is used.

Over the last couple of decades, engineers and scientists invented many approaches to deal with the random input parameters and applied them into design optimization of electromagnetic devices. These approaches are called stochastic design optimization (SDO) (Yu, 2011), and they can be generally divided into four categories.

Worst-case optimization (WCO) considers all the possible results around the design point with a given maximum deviation, and try to make sure that the worst-case in this neighbourhood will not violate constraints (Parkinson, Sorensen, & Pourhassan, 1993). This approach and its variations has been used successfully in the superconducting magnetic energy storage system by (Ren, Pham, & Koh, 2013; Steiner, Weber, & Magele,

2004), and in a magnetic resonance imagining device by (Chiariello, Formisano, Martone, & Pizzo, 2015).

Robust design optimization (RDO) aims at making the optimum performance less sensitive to uncertainties. In other word, the main target is to reduce the variability of the system performance, which is characterized most often by its standard deviation (Kang, 2005). This method is widely used in electromagnetic devices like electric motor (Picheral, 2013) and electromagnetic actuator (Neubert, Kamusella, & Pham, 2010) because of its simplicity of operation and its ability to find solutions that compromise between the mean and variance of the objective.

Reliability-based design optimization (RBDO) focuses on finding an optimum with a given maximum probability of failure which the design must at least meet (Dersjö, 2012). RBDO is the most studied method among all the four categories especially in recent years, with the help of probabilities, the designers could quantify the violation of constraints and provide an improved design with a higher level of confidence. Aerospace and design of vehicles are two of the areas that stimulate this type of optimization (Park, Lee, Chung, & Behdinan, 2015).

Reliability-based robust design optimization (RBRDO) is a method which takes both RBDO and RDO as the research targets (Du & Chen, 1999). In this approach, the robust design is obtained under uncertain constraints quantified by the associated probability of failure. It is the latest method and the applications have not yet spread widely but as a method which absorb both features of RDO and RBDO, many engineers show interest in its applications, and recently (Zhang, 2015) used it for vehicle components.

Each category has various approaches using different manners to achieve their purpose and with these categories of methods, uncertainties in design parameters could be taken into account and get an optimum more robust and/or more feasible. That means the optimum varies little with the uncertainties or has less probabilities to violate the constraints.

Although stochastic strategies are able to find the desired region with quite good probability or small variance, they could not deal with the fine models which have high computational cost, like finite element models of electromagnetic devices. It is because that evaluation of all points during optimization process is insufferable for time consuming black-box models. What is more serious is that the former SDO methods aggravate this problem since they change the problems to more complicate ones.

To handle this problem, surrogate model is commonly used to represent the compute-intensive simulation model. The main idea of surrogate model or so-called meta-model is to determine a continuous function of a set of design variables from a limited amount of available data, so that the outputs of other design points can be predicted rapidly by this continuous function instead of the heavy model itself.

Various approaches to build a meta-model have been proposed in last decades, such as polynomial regression, artificial neural network, kriging and so on. As one of the most widely used approach, kriging meta-model will be used in this manuscript. The main idea of kriging is that the points close to samples should get more weight in the prediction to improve the estimate (Lichtenstern, 2013), it is possible to use the knowledge of only a few samples to approach complex functions.

The most important advantage of meta-model is that it is fast to predict an estimate, so that it can be used in optimization algorithms to replace the time consuming evaluation on heavy models. A straightforward way to combine the meta-model with optimization methods is putting them in series. That means first we choose some design point as samples to evaluate, then a meta-model is built with these samples and responses, at last optimization methods can be applied on the meta-model directly.

However, with this process, the global optimum may not be found if the number of samples are too small since the models are not accurate globally. Therefore, in order to find a more accurate solution, a type of method called adaptive meta-model based optimization was proposed. The most representative one is efficient global optimization (EGO) (Jones, Schonlau, & Welch, 1998) that searches the optimum and refines the meta-model in parallel by adding new samples in the promising region with an infill searching criterion (ISC). The goal of ISC is to choose additional points that offer quick convergence to the global optimum while fitting the exact model over the full range of inputs at the same time (Liu, Han, & Song, 2012).

Many ISCs were proposed, nevertheless, most of them can be applied only with DDO and do not take any uncertainty into account. The methods combining the SDO with adaptive meta-models are still a relatively new subject. This limits the applications of SDO on heavy models in electrical engineering and many other industry field.

So as to expand the application range of SDO, this thesis focuses on adapting different categories of SDO methods with adaptive meta-models for electromagnetic devices. As the formulations of SDO methods are diverse, appropriate ISC and process are not the same. Various approaches are analyzed and the most effective or universal one is selected in each SDO category, then in order to find out the suitable processes for adaptive meta-models, different strategies including the choice of the ISC and the positioning of sample enrichment in the optimization process are investigated in this manuscript for each category of SDO approaches.

This thesis contains four chapters:

Chapter 1 involves the presentation of the aforementioned four SDO categories for fast models. In each category, different manners are introduced in details and a simple mathematic example is used to compare them. Then, a synthesis is made to show their respective merits and drawbacks.

Chapter 2 first introduces the kriging principles, then EGO and the main ISC in application are presented also. The next parts propose some strategies to combine the different SDO with adaptive kriging models to handle the heavy models. They describe which ISC should be used and where the enrichment of samples should be placed. The same mathematic example as in Chapter 1 is also used here to verify the feasibility of the proposed methods and to select the most effective one for each category of SDO.

In Chapter 3, a typically electromagnetic device that is a safety transformer is introduced as a real application to verify the conclusions drawn with the mathematical example. Analytic model and finite element model are both presented and a comparison is made between them. Since analytic model is quick, it is suitable for the methods for fast models. On the other hand, the finite element model is tested with methods for heavy models proposed in Chapter 2. Then a more general conclusion is made on this electromagnetic device optimization.

The last chapter gives conclusions of this work and some prospects for future works.

Chapter 1: Methods adapted to fast models

1 State of the art

Generally speaking, optimization problems contain the input variables, one or more objective functions and the constraints which should be respected as outputs. The formulation of this classic type of optimization is:

$$\begin{aligned} &\min_{\boldsymbol{d}\in S} f(\boldsymbol{d}) \\ &s.t.\ \boldsymbol{g}(\boldsymbol{d}) \leq 0 \\ &\boldsymbol{h}(\boldsymbol{d}) = 0 \end{aligned} \tag{1.1}$$

where $\boldsymbol{d}$ represents the input variables which can be a vector, the search space S is defined by different types of $\boldsymbol{d}$: if the variables are continuous S: $\boldsymbol{d}^L \leq \boldsymbol{d} \leq \boldsymbol{d}^U$, if they are discrete $S = \{a, b, c, \dots\}$. $f(\boldsymbol{d})$ is the objective or cost function, $\boldsymbol{g}(\boldsymbol{d})$ and $\boldsymbol{h}(\boldsymbol{d})$ denote the inequality constraints and equality constraints respectively. As one equality constraint can be also written with two inequality constraints as $-\varepsilon \leq \boldsymbol{h}(\boldsymbol{d}) \leq \varepsilon$ where ε is a small positive tolerance, sometimes only inequality constraints are used to express the problem.

This type of optimization is called deterministic optimization, however the solutions found by this classic type of optimization may not be very interesting in practice for electromagnetic problems, as they do not take into account the uncertainties which are inevitable in industry or other real applications.

The unavoidable uncertainties which can influence the output values are of multiple origins (Beyer & Sendhoff, 2007), including uncertainties due to environmental conditions, inaccuracy on variables, measurement uncertainties on the output of the system. Sometimes a small input variability can lead to a large variability on the output, so that considering the uncertainties during optimization is important to improve the feasibility of the obtained solution.

The uncertainties can be reduced but never eradicated completely, for that reason ways to decrease the influence of uncertainties with existing techniques is quite worth studying. Especially in applications of engineering designs, less influence signifies less cost or more stable performance, which are quite valuable in real world.

With technology evolved in last decades, various methods are proposed to tackle the effects of input uncertainties for design optimization problems. They can be roughly separated into four categories:

- Worst-Case Optimization (WCO)
- Robust Design Optimization (RDO)
- Reliability-Based Design Optimization (RBDO)
- Reliability-Based Robust Design Optimization (RBRDO)

For the first category, WCO, it is a non-probabilistic approach that is based on minimax problem formulation. There is no need to determine the distributions of objective function and constraints. The three others are probabilistic approaches that quantify the uncertainty of quantities of interest by considering inputs and outputs as random variables. RDO should use the distribution of inputs to calculate the statistical moments of objective function and constraints (mean and standard deviation) that are used to create robust objective function and constraints. It should be noticed that only the moments can be calculated by uncertainty propagation techniques, the distribution of objective and constraints could be known only if the model is linear. RBDO needs to analyse the reliability and calculate the probabilities of failure as new constraints.

This chapter will first introduce these disparate categories of optimization with different formulations and then numerical investigation on these methods will be presented.

1.1 Worst Case Optimization (WCO)

A widely used method to evaluate the reliability of solution is the worst case optimization. As shown in Fig. 1.1, A, B, C are the design points, the grey rectangles around them represent the possible performance caused by uncertainties. A small disturbance act on the classical optimum A can lead to an unfeasible solution A'.

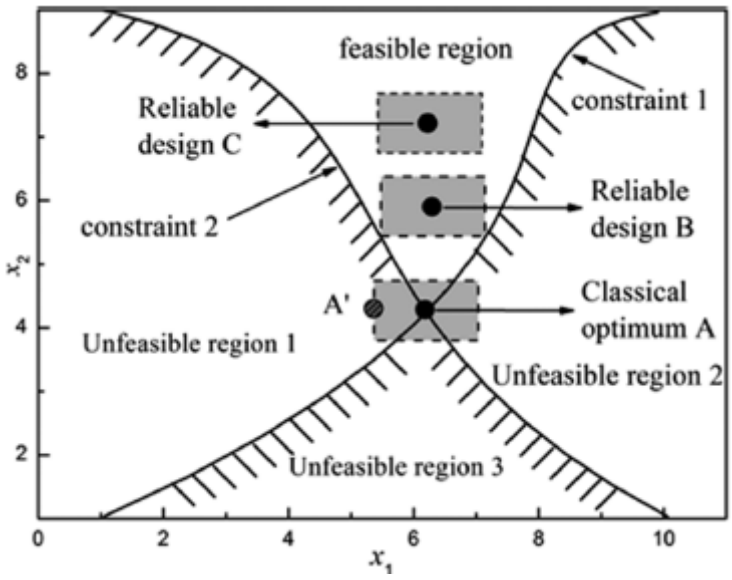


Fig. 1.1. Classical optimum, worst case optimum and reliable design (Song, Li, Rotaru & Sykulski , 2014)

Therefore, the principle of the worst-case optimization technique is that we consider not only the exact design point but also a region around this point as an influence of uncertainties. If any point in this region might enter the unfeasible area, this design will be abandoned. On other words, a design will be chosen as an optimum only if the worst case point in the region we consider do not violate the constraints, like the design B in Fig. 1.1. The uncertainty set can be defined as:

$$U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^p | \boldsymbol{d} - k\boldsymbol{\sigma} \leq \boldsymbol{\xi} \leq \boldsymbol{d} + k\boldsymbol{\sigma}\} \tag{1.2}$$

where $\boldsymbol{\sigma}$ is the standard deviation of uncertain variables with normal distribution, k is determined according to the confidence level (Ren, Pham, & Koh, 2013) which signifies the probability that a random variable lies within the confidence interval of an estimate. The value of k is decided by designers, usually chosen between 1 to 6, larger value corresponds to higher probability, for example, $k = 2$ means that 95.45% of cases should fall within properly-calculated intervals, when $k = 3$, it increases to 99.73%. p is the dimension of input variables. For uncertain variables following uniform law, $\boldsymbol{\sigma}$ is the maximum deviation and k is one.

Another model of uncertainty set is a hyper-ellipsoid that can be suitable for uncorrelated Gaussian uncertainties. In this case, the uncertainties are defined as:

$$U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^p | (\boldsymbol{\xi} - \boldsymbol{d})^T (Q^{-1})^2 (\boldsymbol{\xi} - \boldsymbol{d}) \leq 1\} \tag{1.3}$$

where $Q = k\,\mathrm{diag}(\boldsymbol{\sigma}_i)$ (Steiner, Weber, & Magele, 2004). For the sake of simplicity, the model of hyper-rectangle is chosen as the uncertainty set in this manuscript.

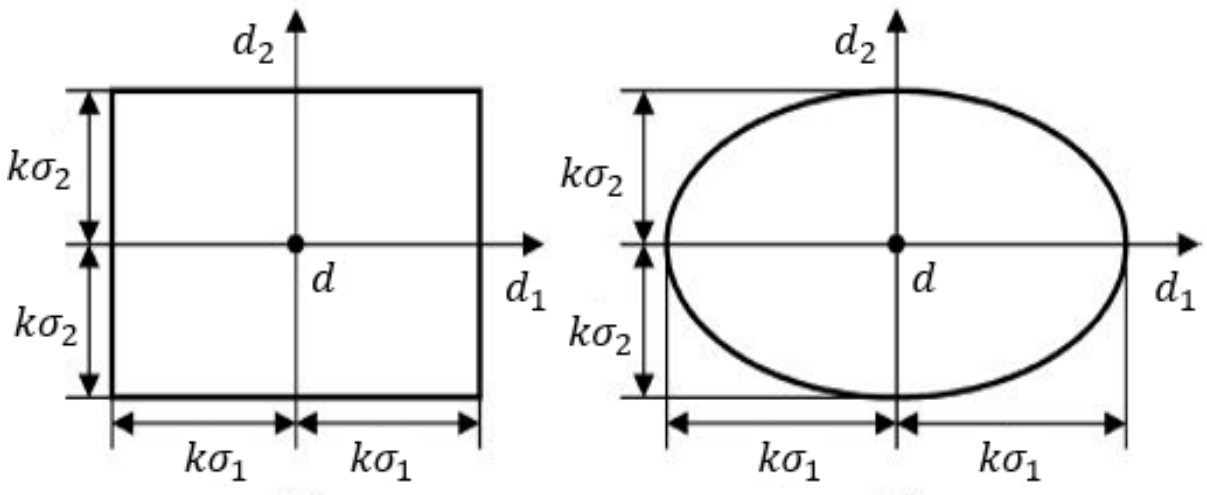


Fig. 1.2. Different uncertainty set (Steiner et al., 2004)

For the purpose of finding the worst scenario considering uncertainties, the WCO method may be expressed as a minimax problem like:

$$\begin{aligned} &\min_{\boldsymbol{d}\in S} f_w(\boldsymbol{d}) \\ &s.t.\ \boldsymbol{g}_w(\boldsymbol{d}) \le 0 \\ \text{with}\quad &f_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi}\in U(\boldsymbol{d})} f(\boldsymbol{\xi}) \\ &\boldsymbol{g}_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi}\in U(\boldsymbol{d})} \boldsymbol{g}(\boldsymbol{\xi}) \end{aligned} \tag{1.4}$$

That means the original values of the design variables $\boldsymbol{d}$ for the objective function and the constraint functions are substituted by the worst values. The process of WCO method is shown in the Fig 1.3.

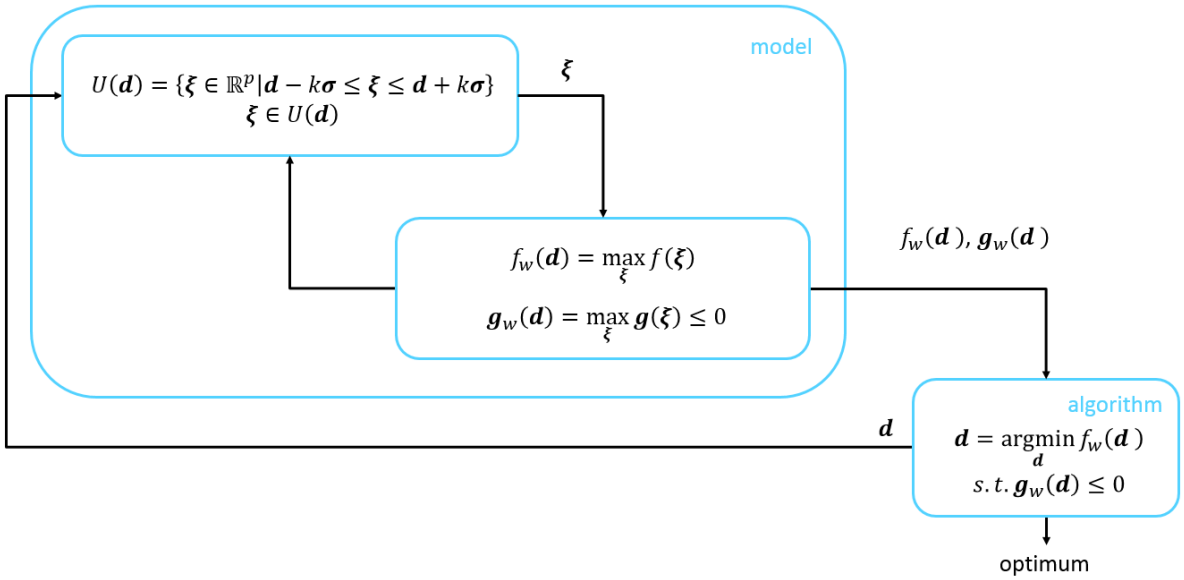


Fig. 1.3. Process of Worst Case Optimization

However, as a minimax optimization, there is one optimization loop nested in another. This time-consuming process slows down the efficiency vastly because the sub-problem has to be solved for every point in the design space found by optimization in each iteration of optimization. In order to reduce the computational burden, (Sundaresan, Ishii, & Houser, 1995) propose a technique to use the vertex discretization to replace the inner maximal optimization. That means the infinite uncertainty set $U(\boldsymbol{d})$ is replaced with the finite set U_v, which consists of only the vertices of the initial $U(\boldsymbol{d})$.

$$U_v = \{\boldsymbol{d} + k\boldsymbol{\sigma}; \boldsymbol{d} - k\boldsymbol{\sigma}\} \tag{1.5}$$

This method is based on a first order Taylor expansion so the vertices are likely to exhibit the worst values in the region $U(\boldsymbol{d})$ because they are the farthest points from the design point $\boldsymbol{d}$, i.e. the center. Therefore, the initial inner optimization whose aim is to find the worst value in the region $U(\boldsymbol{d})$ by solving an optimization problem is substituted by the calculation of the vertex value. This discretization technique can be expanded to hyper-rectangular uncertainty sets with p variables and 2^p vertices to evaluate. As shown in Fig. 1.4, the 8 red points present the vertices for a 3 dimension hyper-rectangular.

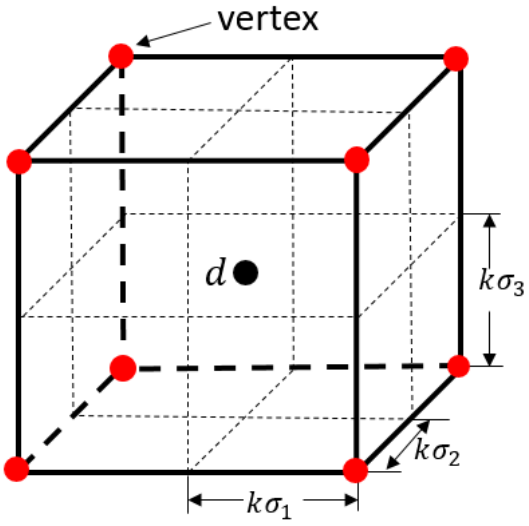


Fig. 1.4. A hyper-rectangular uncertainty set and the corner points for a three-dimensional variables

Although this method avoid a nested optimization problem, 2^p evaluations is still a big disadvantage. Its exponential complexity is a great restriction especially for higher dimensional problems. To reduce the number of evaluations, approximation based WCO methods are proposed. Two representative methods are presented hereafter.

1.1.1 Worst-vertex-based WCO

A worst-vertex-based WCO (WWCO) is proposed by (Steiner et al., 2004). This method aims to observe the values of the bounds of $U(\boldsymbol{d})$ in each dimension and predict the worst vertex value.

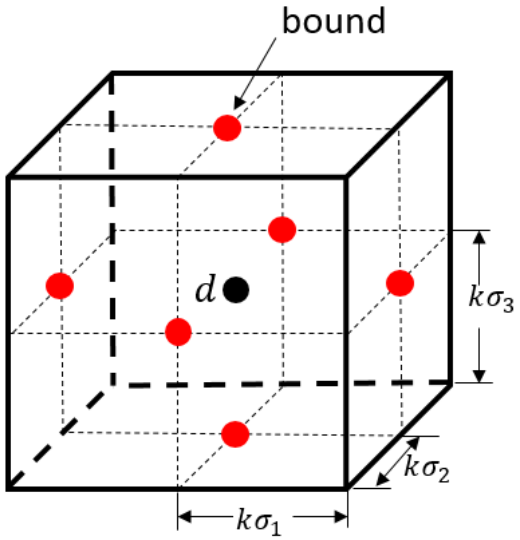


Fig. 1.5. A hyper-rectangular uncertainty set and the bound points for a three-dimensional variables

As shown in Fig. 1.5, the 6 intersections of each two dashed lines on faces are the bounds in each dimension, so that this method needs to evaluate $2p$ points and it is obvious that the number of evaluations for this method will be much less than the one requires by the method based on the calculation at all the vertices when the dimension increases.

The principle of WWCO is to determine the directions of ascent in which the values of objective function and constraints increase (Alotto, Magele, Renhart, Weber, & Steiner, 2003). As in Fig 1.6, the curves are contours of $\bullet\,(\boldsymbol{d})$, white circle points at the upper and lower boundaries of the uncertainty set along each direction i are evaluated. The difference between each two evaluated values at the same dimension shows the directions of ascent in this dimension. It is assumed that the worst vertex $\boldsymbol{d}_{w,\bullet}$ occurs in the direction where the higher value was detected.

$$\boldsymbol{d}_{w,\bullet} = \boldsymbol{d} + \begin{Bmatrix} sign\big(\bullet\,(\boldsymbol{d}+k\sigma_1) - \bullet\,(\boldsymbol{d}-k\sigma_1)\big)\cdot k\sigma_1 \\ \vdots \\ sign\left(\bullet\,(\boldsymbol{d}+k\sigma_p) - \bullet\,(\boldsymbol{d}-k\sigma_p)\right)\cdot k\sigma_p \end{Bmatrix} \tag{1.6}$$

where p is the dimension of variables, $\bullet$ can be the objective function f or constraints $\boldsymbol{g}$.

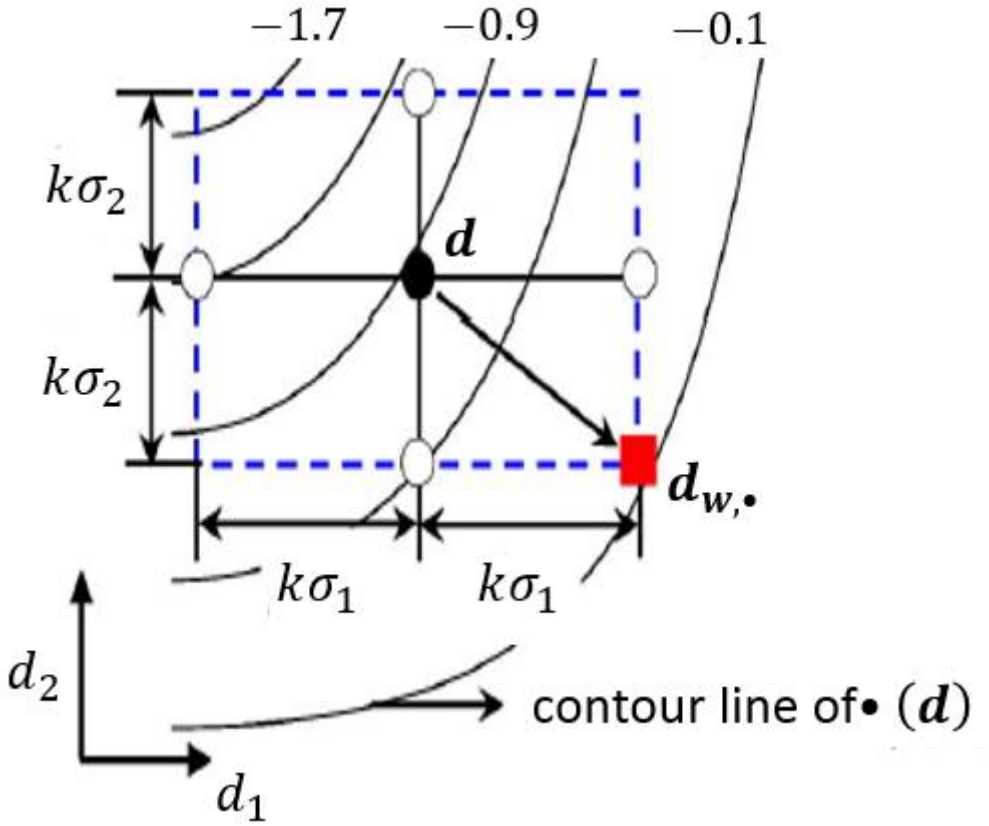


Fig. 1.6. Worst-Vertex-based WCO (Z. Ren et al., 2013)

As the directions of ascent may not always be the same for the objective function and each constraint, so the worst vertex should also be calculated for every constraint.

Therefore, the total number of evaluations is $2p + m + 1$ for a design point in outer optimization loop with m is the number of constraints.

According to these worst vertices, the worst-case objective and constraints can be approximated as:

$$\begin{aligned} \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} f(\boldsymbol{\xi}) &\cong f\left(\boldsymbol{d}_{w,f}\right) \\ \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \boldsymbol{g}(\boldsymbol{\xi}) &\cong \boldsymbol{g}\left(\boldsymbol{d}_{w,g}\right) \end{aligned} \tag{1.7}$$

So, the formulation is changed into:

$$\begin{aligned} &\min_{\boldsymbol{d} \in S} f\left(\boldsymbol{d}_{w,f}\right) \\ &s.t. \quad \boldsymbol{g}\left(\boldsymbol{d}_{w,g}\right) \leq 0 \end{aligned} \tag{1.8}$$

1.1.2 Gradient-based WCO

Another type of method to reduce the computational burden for WCO is gradient-based method (Z. Ren et al., 2013; Z. Ren, Pham, Song, Kim, & Koh, 2011). These methods intend to use Taylor series expansion in the neighborhood of the mean value $\boldsymbol{d}$, if the objective function and constraints are differentiable.

The expansion is:

$$\bullet(\boldsymbol{\xi}) = \bullet(\boldsymbol{d}) + \nabla \bullet(\boldsymbol{d}) \cdot (\boldsymbol{\xi} - \boldsymbol{d}) + \frac{1}{2}(\boldsymbol{\xi} - \boldsymbol{d})^T H(\boldsymbol{d})(\boldsymbol{\xi} - \boldsymbol{d}) + \cdots \quad \forall \boldsymbol{\xi} \in U(\boldsymbol{d}) \tag{1.9}$$

where $\nabla \bullet(\boldsymbol{d})$ is the gradient and $H(\boldsymbol{d})$ is the Hessian matrix of $\bullet(\boldsymbol{d})$, $\bullet$ can be the objective function f or constraints $\boldsymbol{g}$.

For first order gradient-based WCO (GWCO), the higher order terms are neglected and with the help of Cauchy–Schwarz inequality, the objective function values in the region around $\boldsymbol{x}$ can be written as:

$$\begin{aligned} \bullet(\boldsymbol{\xi}) &\approx \bullet(\boldsymbol{d}) + \nabla \bullet(\boldsymbol{d}) \cdot (\boldsymbol{\xi} - \boldsymbol{d}) \leq \bullet(\boldsymbol{d}) + \|\nabla \bullet(\boldsymbol{d})\| \cdot \|\boldsymbol{\xi} - \boldsymbol{d}\| \\ \bullet(\boldsymbol{\xi}) &\leq \bullet(\boldsymbol{d}) + \|\nabla \bullet(\boldsymbol{d})\| \cdot \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \|\boldsymbol{\xi} - \boldsymbol{d}\| \quad \forall \boldsymbol{\xi} \in U(\boldsymbol{d}) \end{aligned} \tag{1.10}$$

So the boundary in right side in (1.10) is treated as the worst case for $f(\boldsymbol{d})$. Same approximation can be introduced into constraints and the initial worst-case problem turns to be an approximation with gradient information:

$$\begin{aligned} &\min_{\boldsymbol{d} \in S} f_w(\boldsymbol{d}) \\ &s.t. \ \boldsymbol{g}_w(\boldsymbol{d}) \leq 0 \\ with \quad &f_w(\boldsymbol{d}) = f(\boldsymbol{d}) + \|\nabla f(\boldsymbol{d})\| \cdot \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \|\boldsymbol{\xi} - \boldsymbol{d}\| \\ &\boldsymbol{g}_w(\boldsymbol{d}) = \boldsymbol{g}(\boldsymbol{d}) + \|\nabla \boldsymbol{g}(\boldsymbol{d})\| \cdot \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \|\boldsymbol{\xi} - \boldsymbol{d}\| \end{aligned} \tag{1.11}$$

The principle of second order gradient-based WCO is almost the same, objective function and constraints are approximated with a second order Taylor expansion:

$$f_{so}(\boldsymbol{\xi}) \approx f(\boldsymbol{d}) + \nabla f(\boldsymbol{d}) \cdot (\boldsymbol{\xi} - \boldsymbol{d}) + \frac{1}{2}(\boldsymbol{\xi} - \boldsymbol{d})^T H(\boldsymbol{d})(\boldsymbol{\xi} - \boldsymbol{d}) \tag{1.12}$$

Suppose that the worst performance $f_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \left(f_{so}(\boldsymbol{\xi})\right)$, therefore the second order WCO is formulated as:

$$\begin{aligned} &\min_{\boldsymbol{d} \in S} f_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \left(f_{so}(\boldsymbol{\xi})\right) \\ &s.t. \quad \boldsymbol{g}_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \left(\boldsymbol{g}_{so}(\boldsymbol{\xi})\right) \leq 0 \end{aligned} \tag{1.13}$$

Normally in engineering applications, it is hard to calculate the derivatives of orders higher than one, for that reason, the following parts will only consider the first order gradient-based WCO.

1.2 Robust Design Optimization (RDO)

It is evident that deterministic optimization without considering the uncertainties of input parameters will find a global optimum that lies on one or several constraints boundaries in most cases. With a small deviation on the solution, this one could easily violate one or more constraints and fall into the failure domain. Moreover, if the global optimum lies on a very narrow valley of objective function, even a slight variation in the variables could result in a significant change for the performance. In order to reduce the impact of disturbance and improve the robustness, RDO method is proposed.

Within the RDO process, the statistical variability of the design parameter is considered and the original deterministic input parameter $\boldsymbol{d}$ is replaced by a random input parameter $\boldsymbol{X}$. For the sake of simplicity and without loss of generality, in this manuscript all the input variables with uncertainty follow a Gaussian law where the mean value is denoted $\boldsymbol{d}$ and the standard deviation is denoted $\boldsymbol{\sigma}$ which is considered constant.

Optimized design within the sigma level $\pm k\boldsymbol{\sigma}$ is characterized as robust design. The objective of the RDO is to find a design with a minimal variance of the model responses around the mean values of the design parameters (Roos, Adam, & Bucher, 2006). As shown in Fig. 1.7, $\boldsymbol{d}_{min}$ is the global optimum which can be obtained by deterministic optimization. However, the variance between $\pm k\boldsymbol{\sigma}$ around this point in the input variable space result in larger variance Δy in the output variable space than the one around the local optimum $\boldsymbol{d}'_{min}$. Thus, $\boldsymbol{d}'_{min}$ is more robust than $\boldsymbol{d}_{min}$.

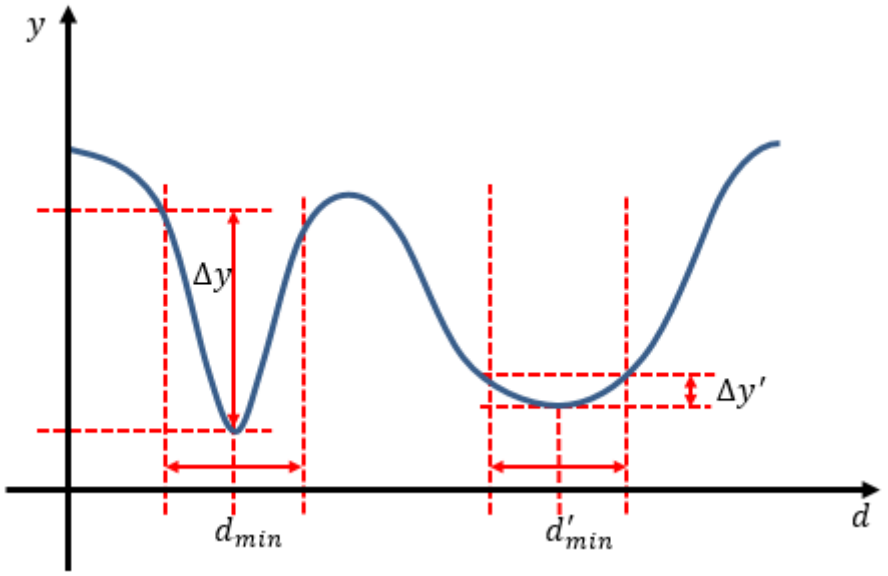


Fig. 1.7. Deterministic optimum and robust optimum

Therefore, a new robust problem involves not only the objective function itself but also the variance of performance. So that it would be reformulated as:

$$\begin{cases} \min\limits_{\boldsymbol{d}} f_r\left(\mu_f(\boldsymbol{d}), \sigma_f(\boldsymbol{d})\right) \\ s.t.\ \ \boldsymbol{g_r}\left(\boldsymbol{\mu_g}(\boldsymbol{d}), \boldsymbol{\sigma_g}(\boldsymbol{d})\right) \le 0 \\ \boldsymbol{d}^L + k\boldsymbol{\sigma} \le \boldsymbol{d} \le \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{cases} \tag{1.14}$$

where f_r and g_r are the new robust objective function and constraints which normally contains both the mean $\mu_f(\boldsymbol{d}), \boldsymbol{\mu_g}(\boldsymbol{d})$ and standard deviation $\sigma_f(\boldsymbol{d}), \boldsymbol{\sigma_g}(\boldsymbol{d})$ of initial ones. Besides, the lower bound and upper bound of the robust problem are modified also. In deterministic problems, the constraint of bound is $\boldsymbol{d}^L \le \boldsymbol{d} \le \boldsymbol{d}^U$, but when we consider uncertainties and the input variables follow the normal law $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{d}, \boldsymbol{\sigma}^2)$, in order to make sure the value of $\boldsymbol{X}$ is in the range, new bounds should be as $\boldsymbol{d} - k\boldsymbol{\sigma} \ge \boldsymbol{d}^L$ and $\boldsymbol{d} + k\boldsymbol{\sigma} \ge \boldsymbol{d}^U$, where k is the same confidence level as in WCO. To find the minimal variance of objective and constraints, their means μ_f, μ_g and standards deviations σ_f, σ_g should be calculated first. They may be computed as the following equations (Paiva, 2010).

$$\begin{cases} \mu_f(\boldsymbol{d}) = \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} f(t) f_X(t) dt \\ \sigma_f(\boldsymbol{d}) = \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} \left[f(t) - \mu_f(d)\right]^2 f_X(t) dt \end{cases} \tag{1.15}$$

where f represents the initial function objective, t has the same dimension with $\boldsymbol{X}$, f_X is the joint probability density function of $\boldsymbol{X}$. The mean and standard deviation of constraints g can be calculated in the same way. However, the analytical evaluation of the integrals in (1.15) seems impossible to compute in most practical problems, for that reason, some numerical methods are proposed to obtain a quite good approximate result of these moments.

The techniques can be separated roughly into three groups, the first is the simulation methods, and the most general method in this group is the well-established Monte Carlo simulation with a huge quantity of samples. The second group is the perturbation methods and the most typical one is Taylor based Method of Moments (Padulo, Forth, & Guenov, 2008). The last one is approximation methods for instance by using chaos polynomials. The former two type of methods will be used in this manuscript and the following sections will show more details of the mentioned typical methods: Monte Carlo simulation and Taylor based Method of Moments.

1.2.1 Monte Carlo simulation

Monte Carlo method is a simulation method commonly used in particle physics, or it can also introduce a statistical approach to assess risk in a finance. The purpose of the Monte Carlo simulation is to estimate quantities like mean, PDF… of the output parameters when the input parameters of the model are random variables. It is a numerical method for statistical simulation that uses sequences of random numbers to perform the simulation.

The approach choose n values for each input parameter $X^{(i)}(i = 1, \dots, n)$ randomly (n should be very large) depend on its probability distribution (Amelin & Kungliga tekniska högskolan (Stockholm), 2004). The model is then computed for each sample set, that means evaluated n times and then for each output parameter a vector with n values is obtained. With these values, quantities of interest can be estimated like the moments of the output parameters.

A simple example is used here to describe the principle of Monte Carlo method. Given an integral:

$$\int_{[d^L, d^U]} J(x)dx \tag{1.16}$$

where $J(x)$ is a real integrable function and X is a random variable within the interval $[d^L, d^U]$. Before presenting the distribution (mean and standard deviation) of the output of this integral, some fundamental laws should be introduced.

The first one is the law of large numbers:

If $X^{(1)}, X^{(2)}, \dots, X^{(n)}$ are independent and identically distributed (iid), the expectation exist and $E\left[X^{(1)}\right] = E\left[X^{(2)}\right] = \cdots = E\left[X^{(n)}\right] = \mu$, then

$$P\left(\lim_{n\to\infty} \bar{X}_n = \mu\right) = 1 \tag{1.17}$$

where

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X^{(i)} \tag{1.18}$$

That means if sample number n goes to infinity, the probability that the average of the observations is equal to the expected value will 1, in this condition $\bar{X}_n$ can be used as an estimator of $E[X]$.

The second law is the central limit theorem:

Let $X^{(1)}, X^{(2)}, \dots, X^{(n)}$ be a sequence of random variables that are iid, and $E(X^2) < \infty$ exist with $E(Y) = \mu_Y$ and the variance is σ_Y^2 for $i = 1,2, \dots$, it is noted that

$$S_n = \frac{1}{n}\sum_{i=1}^{n} x^{(i)} \tag{1.19}$$

also, $x^{(i)}$ is a realisation of $X^{(i)}$. The empirical mean S_n has a normal distribution $\mathcal{N}(\mu_X, \sigma_X^2/n)$ if n is large enough.

With the two fundamental laws, the distribution of output and also the rate of convergence can be obtained. Note that in the example above, the mean and standard deviation can be calculated by the following equation:

$$\begin{cases} \mu_J \approx \frac{1}{n}\sum_{i=1}^{n} J\left(x^{(i)}\right) \\ \sigma_J^2 \approx \frac{1}{n-1}\sum_{i=1}^{n}\left(J\left(x^{(i)}\right) - \mu_J\right)^2 \end{cases} \tag{1.20}$$

where $J\left(x^{(i)}\right)$ is the i-th value of the output parameters, μ_J and σ_J are the mean and standard deviation of the output parameters respectively.

However, the major shortcoming of this approach is its vast need of computational resources due to a poor convergence rate ($\sim 1/\mathrm{n}^{1/2}$) so requiring a huge number of runs, and these may cost huge time especially for heavy engineering designs.

1.2.2 Taylor Based Method of Moments

Another common approach is based on a Taylor expansion to propagate the uncertainty from input parameters $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{d}, \boldsymbol{\sigma}^2)$ to output parameters $J(\boldsymbol{X})$.

This method consist in relating the moments of output parameters analytically as a function of the moments of the input parameters (Lei, Lima-Filho, Styblinski, & Singh, 1998). The expressions of the output moments are based on reformulations of the model equations using first or second order Taylor expansions.

Given p independent random variables, the first order Taylor expansion for a function $y = J\left(x_1, x_2, \dots, x_p\right)$ around the point $\mu_X = \left[\mu_{X_1}, \mu_{X_2}, \dots, \mu_{X_p}\right]$ which presents the mean of input X is as follows:

$$\tilde{y} = J(\mu_X) + \sum_{i=1}^{p} \frac{\partial J(\mu_X)}{\partial X_i}\left(x_i - \mu_{X_i}\right) \tag{1.21}$$

where $\partial J/\partial x_i$ is the partial derivative of the function J with respect to the variables x_i.

With the help of (1.21), the expressions for the mean and standard deviation of $J(X)$ can be expressed as:

$$\begin{cases} \hat{\mu}_J = J(\mu_X) \\ \hat{\sigma}_J^2 = \sum_{i=1}^{p} \left(\frac{\partial J(\mu_X)}{\partial x_i}\right)^2 \sigma_{X_i}^2 \end{cases} \tag{1.22}$$

The first order Taylor expansion is a linear approximation that is more convenient and only use the first derivative information. However, it is a bit rough when sometimes performance linear approximation is not sufficiently accurate. To solve this problem, a quadratic approximation – second order Taylor expansion is proposed.

As previously, the second order development is expressed as follows:

$$\tilde{y} = J(\mu_X) + \sum_{i=1}^{n} \frac{\partial J(\mu_X)}{\partial x_i}\left(x_i - \mu_{X_i}\right) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 J(\mu_X)}{\partial x_i \partial x_j}\left(x_i - \mu_{X_i}\right)\left(x_j - \mu_{X_j}\right) \tag{1.23}$$

where $\partial^2 J/\left(\partial x_i \partial x_j\right)$ is the second partial derivative of the function J with respect to the variables x_i and x_j.

The moments can be calculated as (Padulo et al., 2008):

$$\begin{cases} \hat{\mu}_J = J(\mu_X) + \frac{1}{2}\sum_{i=1}^{n} \frac{\partial^2 J(\mu_X)}{\partial x_i^2} \sigma_{X_i}^2 \\ \hat{\sigma}_J^2 = \sum_{i=1}^{n} \left(\frac{\partial J(\mu_X)}{\partial x_i}\right)^2 \sigma_{X_i}^2 + \frac{1}{2}\sum_{i=1}^{n} \left(\frac{\partial^2 J(\mu_X)}{\partial x_i^2}\right)^2 \sigma_{X_i}^4 + \sum_{i<j}^{n} \left(\frac{\partial^2 J(\mu_X)}{\partial x_i \partial x_j}\right)^2 \sigma_{X_i}^2 \sigma_{X_j}^2 \end{cases} \tag{1.24}$$

Engineering problems involve often nonlinear models, which suggests that the quadratic approximation should be more appropriate for calculating the moments of output (Glancy, 1999). However, it uses more information like second order partial derivatives than the linear approximation and it is usually quite difficult to calculate (Z. Ren et al., 2013) and time-consuming so that the first order Taylor based method is applied in practice in engineering design and especially for the finite element models.

1.2.3 Robust formulations

After the propagation of uncertainty, the moments of outputs are obtained and it is possible to express a robust formulation based on these moments. There are various formulations expressed with the mean and standard deviation of objective and/or constraints. The simplest formulation uses the means of objective function $\mu_f(\boldsymbol{d})$ and constraints $\mu_g(\boldsymbol{d})$ to replace the initial ones (Sundaresan, Ishii & Houser, 1992):

$$\begin{gathered} \min_{\boldsymbol{d}} \mu_f(\boldsymbol{d}) \\ s.t.\ \mu_g(\boldsymbol{d}) \leq 0 \\ \boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{gathered} \tag{1.25}$$

However, this formulation does not take the variance of inputs into account and the robustness cannot be assessed. For that reason, (Fonseca, 2003) adds the standard deviation of objective function as a new robust objective, leading to a bi-objective problem:

$$\begin{gathered} \min_{\boldsymbol{d}} \begin{bmatrix} \mu_f(\boldsymbol{d}) \\ \sigma_f(\boldsymbol{d}) \end{bmatrix} \\ s.t.\ \mu_g(\boldsymbol{d}) \leq 0 \\ \boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{gathered} \tag{1.26}$$

This formulation incorporate the variance as an objective, (K. Deb & Gupta, 2006; K. Deb & Gupta, 2005) offer another way to treat the performance robustness by using an additional constraint:

$$\begin{gathered} \min_{\boldsymbol{d}} \mu_f(\boldsymbol{d}) \\ s.t. \begin{cases} \frac{\|{\mu_f}^w(\boldsymbol{d}) - \mu_f(\boldsymbol{d})\|}{\|\mu_f(\boldsymbol{d})\|} \leq \eta \\ \mu_g(\boldsymbol{d}) \leq 0 \end{cases} \\ \boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{gathered} \tag{1.27}$$

This formulation restricts a normalized change in perturbed objective vector from its original objective vector by a user-specified limit η. The perturbed objective vector ${\mu_f}^w(\boldsymbol{d})$ is the worst mean value of objective function among the uncertainty set $U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^p | \boldsymbol{d} - k\boldsymbol{\sigma} \leq \boldsymbol{\xi} \leq \boldsymbol{d} + k\boldsymbol{\sigma}\}$ which is the neighborhood of design variable like in WCO.

However, the former formulations just consider the performance robustness of objective function and there is no change for constraints. That may cause the optimal solution to lie on the constraint boundaries still and it cannot avoid the optimum from being in the failure domain if there is a deviation. (Asafuddoula, Singh, & Ray, 2015; Picheral, 2013) present other formulations that incorporate the robustness into both objective and constraints.

(K.-H. Lee & Park, 2001) suggest a new robust objective function based on a weighted sum of the mean μ_f and standard deviation σ_f. This formulation transfers the multi-objective problem into a mono-objective one. In addition, the feasibility robustness is expressed as a new constraint:

$$\begin{gathered} \min_{\boldsymbol{d}} \omega \frac{\mu_f(\boldsymbol{d})}{\mu_{f_0}} + (1-\omega) \frac{\sigma_f(\boldsymbol{d})}{\sigma_{f_0}} \\ s.t.\ g(\boldsymbol{d}) + b \sum_{i=1}^{p} \left| \frac{\partial g}{\partial x_i} \right| k\sigma_i \leq 0 \\ \boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{gathered} \tag{1.28}$$

where ω is the weight between 0 and 1. A value close to 0 means that the minimization of the standard deviation is more considered than the mean value, and vice versa. b is the penalty factor and the larger value will enhance the feasibility. μ_{f_0} and σ_{f_0} are two values used for scaling to ensure that the two parts have the same order of magnitude. $\partial g / \partial x_i$ is the first order derivative relative to input parameter x_i. Unlike Equation (1.27), this

formulation change the original constraint with objective at the same time. The limit-state of the reformulated constraints is a translation in the direction of gradient of the initial one in order to increase the feasibility robustness and avoiding the optimal solution lies on a constraint boundary.

(Luo, Wang & Du, 2012) propose another formulation that combines μ_f and σ_f with weights:

$$\begin{aligned} &\min_{\boldsymbol{d}} \omega_1 \mu_f(\boldsymbol{d}) + \omega_2 \sigma_f(\boldsymbol{d}) \\ &s.t.\ \boldsymbol{\mu_g}(\boldsymbol{d}) + k\boldsymbol{\sigma_g}(\boldsymbol{d}) \leq 0 \\ &\boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{aligned} \tag{1.29}$$

where ω_1 and ω_2 are the weights given to the mean and standard deviation respectively. Here the transformation of constraints has the same principal with in Equation (1.28) but it uses the standard deviation of constraints directly as it contains the information of gradients also as can be seen from Equation (1.22).

However, the mean and the standard deviation are not scaled as in the previous formulation. Thus, the value of the mean, generally larger than that of the standard deviation, can create an imbalance within the objective function.

(Wang, Huang, & Liu, 2010; Xinying Liu et al., 2008) use $k\sigma_f$ to formulate the robust objective in the same formulation as constraints:

$$\begin{aligned} &\min_{\boldsymbol{d}} \mu_f(\boldsymbol{d}) + k\sigma_f(\boldsymbol{d}) \\ &s.t.\ \boldsymbol{\mu_g}(\boldsymbol{d}) + k\boldsymbol{\sigma_g}(\boldsymbol{d}) \leq 0 \\ &\boldsymbol{d}^L + k\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - k\boldsymbol{\sigma} \end{aligned} \tag{1.30}$$

The expression $\mu_f + k\sigma_f$ can be considered as an upper bound of the probability distribution of the objective function. The probability that the value of f is less than this bound depends on the value of parameter k. For example, if k is chosen as 3, at least 99.73% of the values will be less than this limit. Thus minimizing $\mu_f + k\sigma_f$ is equivalent to minimizing the worst possible value of the objective function f.

This form that takes both performances feasibility and robustness into account seems to be more practical and widely adopted.

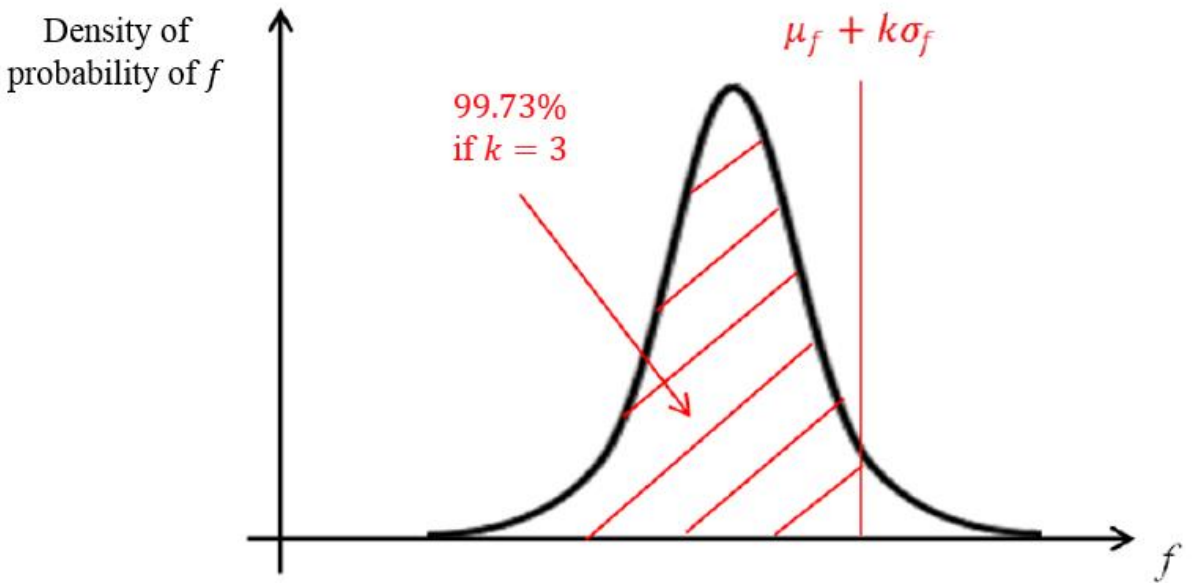


Fig. 1.8. Principe of $\mu_f + k\sigma_f$ (Picheral, 2013)

The aim of RDO is to find the mean value of design variables within a feasible space leading to lower sensitivity of the objective and constraints to uncertainty on design variables. The process for all these formulations are almost the same: Firstly, the input parameters are modelled by their distributions. Secondly, for the reason of simplicity and less time consuming, the Taylor Based Method of Moments is used to propagate the uncertainty from input to output parameters and calculate the statistical moments of the objective function and constraints. At last, robust formulations are used with optimization algorithms to solve the new robust problem.

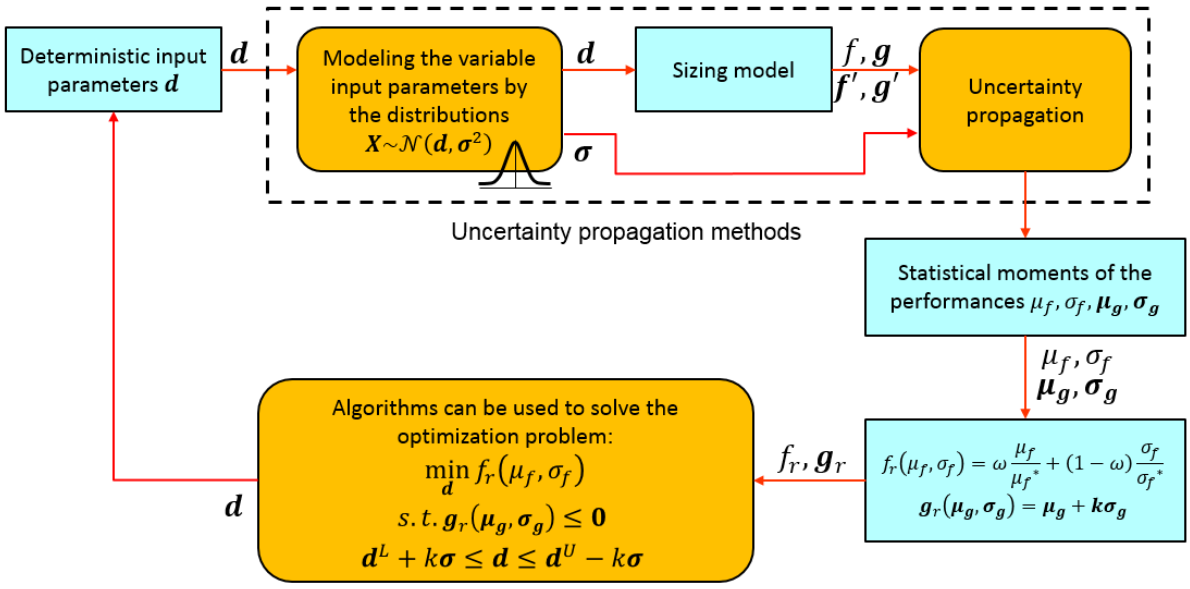


Fig. 1.9. Process of Robust Design Optimization

1.3 Reliability-Based Design Optimization (RBDO)

The method RBDO aims to find the optimal design which satisfies a given targeted reliability level represented by probability of failure. In sampling method (Hit or Miss), the probability of failure is related to the number of points in a sampling around the mean value that fall outside of the reliable domain. This means that RBDO attempts to find the optimal design in order to have the probability of failure smaller than a given targeted value. The formulation of RBDO is as follows:

$$\begin{gathered} \min_{\boldsymbol{d}} f(\boldsymbol{d}) \\ s.t. \quad P_f \leq P_t \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.31}$$

where P_f is the probability of failure, β_t is the targeted value for the reliability index, and $P_t = \Phi(-\beta_t)$ is the targeted value for the probability of failure. RBDO uses probabilistic constraints to make sure that the design variables satisfy a desired reliability level while minimizing the deterministic objective function.

For a better understanding of RBDO principle, the definition of probability of failure and its calculation should be presented first.

1.3.1 Probability of failure

The probability is the likelihood of an event, estimated by a real number between 0 and 1. The Probability Density Function (PDF) and the Cumulative Distribution Function (CDF) define the occurrence of stochastic quantity inherently uncertain. In mathematics, the PDF of a continuous random variable is a function that describes the likelihood of an output value of this random variable near a certain point. CDF can fully describe the probability distribution of a real random variable, and it is the integral of the PDF. The statistical description of a random variable X given by its CDF F_X or PDF f_X is expressed as follows:

$$P[X \leq x] = F_X(x) = \int_{-\infty}^{x} f_X(\tau) d\tau \tag{1.32}$$

where P is the probability of occurrence of an event.

In the field of electromagnetic device manufacturing, the ability to satisfy consumer's demand or operating constraints is important. The reliability means that designers should reduce the probability of failure as much as possible. The determination of the reliability of the device is based on the limit-state function. Each constraint $g(\boldsymbol{X}) \leq 0$ can separate the domain of X into three parts: the limit-state curve is $g(\boldsymbol{X}) = 0$, the domain where $g(\boldsymbol{X}) > 0$ is the failure domain, on the contrary, the security domain represents the area $g(\boldsymbol{X}) < 0$. The designers should try to keep the optimum and its "surrounding" (to account for uncertainties) from the failure domain as far as possible or at least reach a targeted probability of failure.

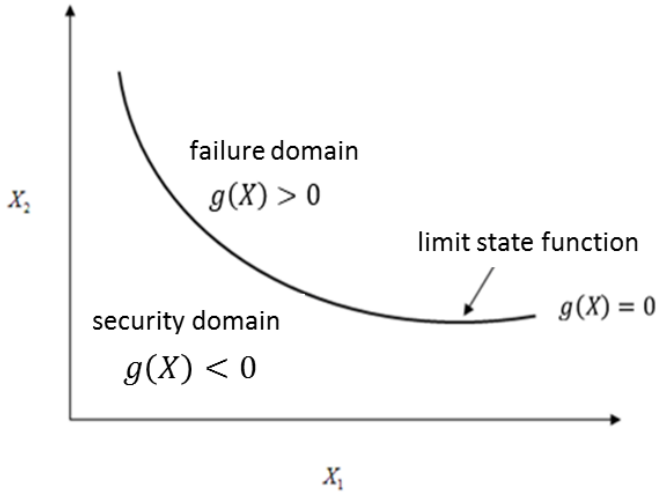


Fig. 1.10. The failure domain, security domain, and limit-state curve

The probability of failure is the probability of the event $g(\boldsymbol{X}) > 0$. It is calculated with the integral:

$$P_f = P[g(\boldsymbol{X}) > 0] = P\left[\boldsymbol{X} \in D_f\right] = \int_{D_f} f_X(\boldsymbol{x}) d\boldsymbol{x} \tag{1.33}$$

where D_f is the failure domain. As the computational burden is heavy with numerous random parameters and complex shape of the failure domain, direct integration is almost impossible and thus Monte-Carlo Simulation (MCS) or other techniques such as First-Order Reliability Method (FORM) is often used to calculate an approximation of P_f (Hasofer & Lind, 1974; Liu & Der Kiureghian, 1991).

MCS is presented in section 1.2.1, and uses a huge number of samples to approach the real probability. It is often said that the number of runs should be hundred times the inverse of the targeted probability of failure (if P_f =10^{-4} then the number of runs should be of 10^6).

FORM and inverse FORM are based on an iso-probalistic transformation to have a normalized vector of statistically independent random variables U instead of the initial input parameter X.

For the Gaussian vector X in this manuscript, the transformation T is as follows:

$$\boldsymbol{T}: \boldsymbol{U} = (\boldsymbol{X} - \boldsymbol{d})/\boldsymbol{\sigma} \tag{1.34}$$

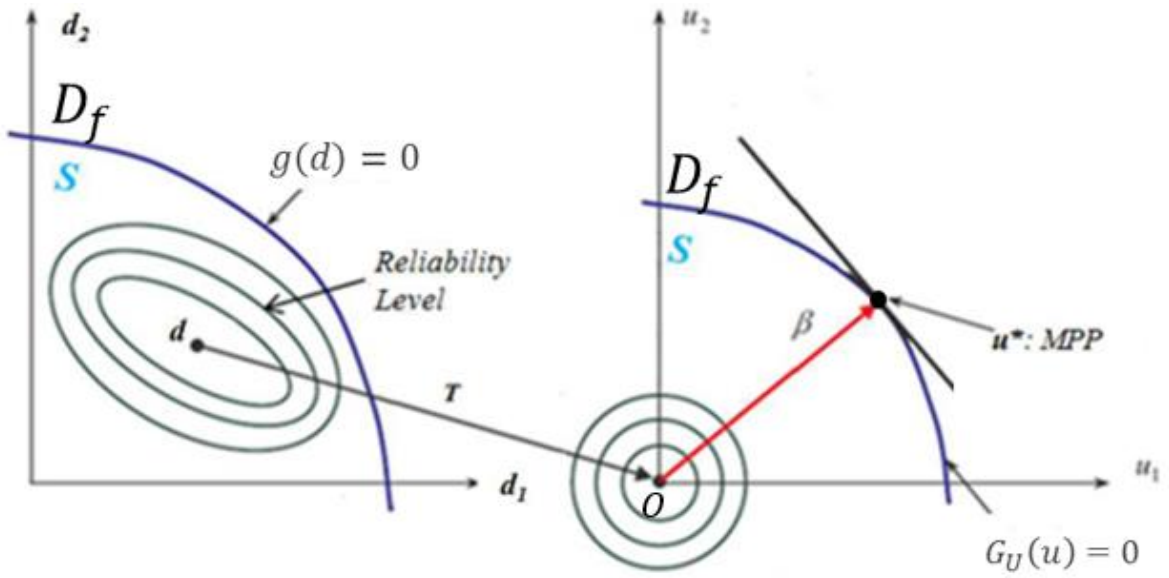


Fig. 1.11. The first order reliability method (Lopez & Beck, 2012)

Then the limit-state function changes from $g(d) = 0$ to $G_U(u) = 0$ and the CDF from F_X to F_U. $G_U(u)$ is defined as the performance function, and the FORM or inverse FORM uses a linear approximation to replace the real $G_U(u)$ at the Most Probable Point of failure (MPP) in U-space. The MPP u^* is the one that minimizes the distance between the origin O and the limit-state $G_U(u) = 0$. After u^* is found, the limit-state function is replaced by a tangent hyperplane crossing u^*. So the probability of failure is:

$$P_f \approx \int_{\tilde{G}_U(\boldsymbol{u})>0} f_U(\boldsymbol{u})du = \Phi(-\beta) \tag{1.35}$$

where $\tilde{G}_U(\boldsymbol{u}) = 0$ is the hyperplane that approximates the limit-state function $G_U(\boldsymbol{u}) = 0$, Φ is the standard Gaussian CDF with $\Phi(\boldsymbol{u}) = \frac{1}{\sqrt{2}}\int_{-\infty}^{u} e^{-\frac{u^2}{2}}du$, and $\beta = \|\boldsymbol{u}\|$ is the reliability index which is equal the norm of u. This method can also be applied for some other kind of distribution, the transformations are given in the following table.

Table 1.1. Transformations from u-space to x-space

Distribution type	Transformation
Normal (μ, σ)	$\mu + \sigma u$
Lognormal (μ, σ)	$e^{\mu+\sigma u}$
Weibull (λ, k)	$\lambda[-\ln(\Phi(-u)^a)]^{\frac{1}{k}}$
Uniform (a, b)	$a + (b-a)\left(0.5 + 0.5erf\left(u\sqrt{2}\right)\right)$
Gumbel (v, α)	$v - \frac{1}{\alpha}\ln[-\ln(\Phi(u))]$
Gamma (a, b)	$ab\left(u\frac{1}{\sqrt{9a}} + 1 - \frac{1}{9a}\right)^3$

The formulation (1.31) is the basis of all RBDO methods and the process of RBDO is shown in Fig 1.12.

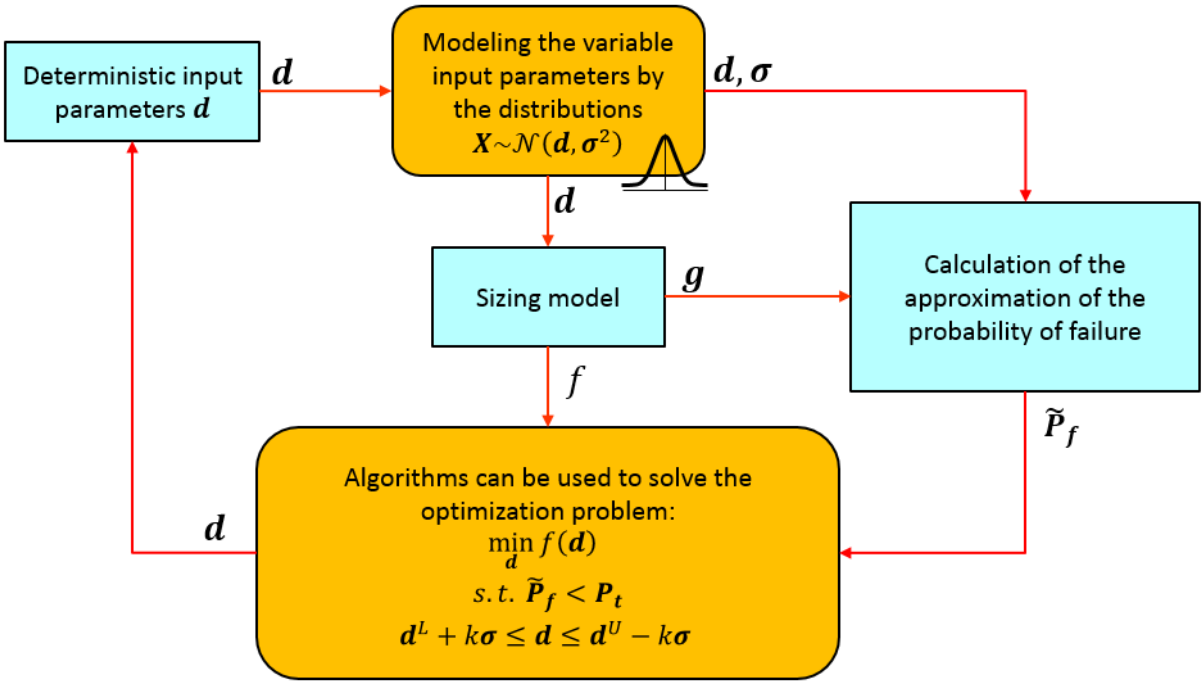


Fig. 1.12. Process of Reliability-Based Design Optimization

How to treat the probabilistic constraints is the main point in RBDO formulations, as it requires a considerable computation effort and it is related to the accuracy and stability of RBDO problem (Aoues & Chateauneuf, 2010). To overcome the numerical difficulties, many approaches have been developed in the past several decades. Usually they can be separated in three main categories:

- double-loop methods
- single-loop methods
- sequential decoupled methods

In the following part, the basic ideas of these categories will be briefly presented and some representative approaches for each category will be detailed.

1.3.2 Double-loop method

Double-loop methods use two loops nested to solve RBDO problems: The inner loop aims to analyse reliability of the current design and to calculate the probability of failure using FORM or inverse FORM. The outer loop seeks the mean values of the design inputs that minimize the objective function and satisfy the probabilistic constraints computed by inner loop (Enevoldsen, 1994). As can be seen from Fig 1.13, another optimization problem is solved to assess the probability of failure.

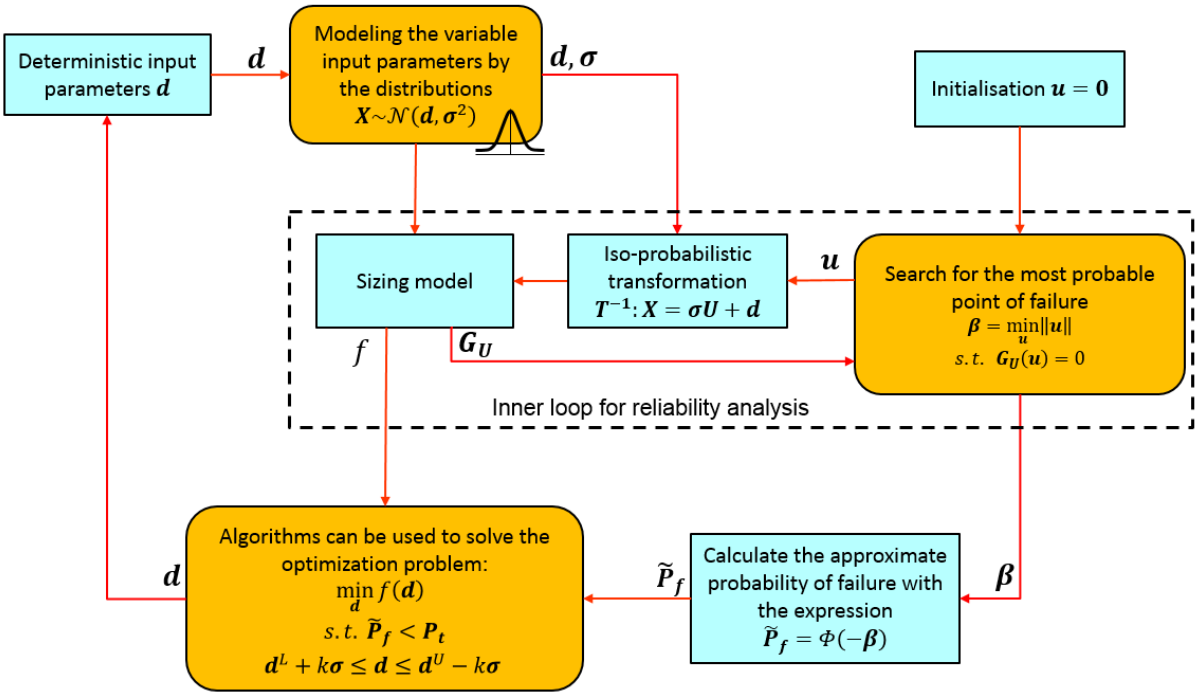


Fig. 1.13. Process of a typical Double-loop method – Reliability Index Approach

In order to solve the nested optimization loops, (Nikolaidis & Burdisso, 1988) proposed a Reliability Index Approach (RIA) based on FORM. It tries to find the MPP for each limit-state during inner loop, then with FORM, the related probability of failure is used as the constraint in outer loop. The MPP can be evaluated by either iterative procedures or simulation methods. Based on this technique, many approximations have been proposed to reduce the computation burden.

(Reddy, Grandhi, & Hopkins, 1994) proposed a simplified safety index based on the advanced second-moment method. (Li & Yang, 1994) decided to use linear programming optimization and linearized reliability index to solve RBDO problems. (Grandhi & Wang, 1998) create an adaptive nonlinear approximation constructed by the function values and the first-order gradient at two points on the limit-state to replace the real limit-state function.

Another classical method of double loop is Performance Measure Approach (PMA). (Tu, Choi, & Park, 1999) uses an inverse reliability problem searching for the maximum performance called Maximum Performance Target Point (MPTP) on the targeted reliability surface to compute the probability.

The principles of RIA and PMA will be introduced in details in the following parts. But before that, the main drawback of the Double loop method should be notice here: it is well recognized that the computation cost for the inner reliability loop can be largely reduce by using approximate method, however this method is still limited due to its inherent two loops.

RIA uses the FORM to calculate the reliability index in the inner loop:

$$\begin{aligned} \beta &= \min_{\boldsymbol{u}} \|\boldsymbol{u}\| \\ s.t.\ \ & G_U(\boldsymbol{u}) = 0 \end{aligned} \tag{1.36}$$

As already presented in Fig 1.11, the minimum on the failure surface is the MPP, to find this solution, either a general optimization algorithm or the HL-RF method can be used (Hasofer & Lind, 1974; Rackwitz & Flessler, 1978; Liu & Der Kiureghian, 1991). The iterative algorithm of HL-RF is:

$$\boldsymbol{u}^{(k+1)} = \left(\boldsymbol{u}^{(k)} \cdot \boldsymbol{n}^{(k)}\right)\boldsymbol{n}^{(k)} - \frac{G_U(\boldsymbol{u}^{(k)})}{\|\nabla G_U(\boldsymbol{u}^{(k)})\|}\boldsymbol{n}^{(k)} \tag{1.37}$$

where k is the number of iteration, $\boldsymbol{n}^{(k)} = \nabla G_U(\boldsymbol{u}^{(k)})/\|\nabla G_U(\boldsymbol{u}^{(k)})\|$ is the ascent direction of the performance function $G_U(\boldsymbol{u})$ at the point $\boldsymbol{u}^{(k)}$. The first term on the right hand side intends to find a direction with the shortest distance to the limit-state and the second term is a correction to reach the performance function (Byeng D. Youn & Choi, 2004a).

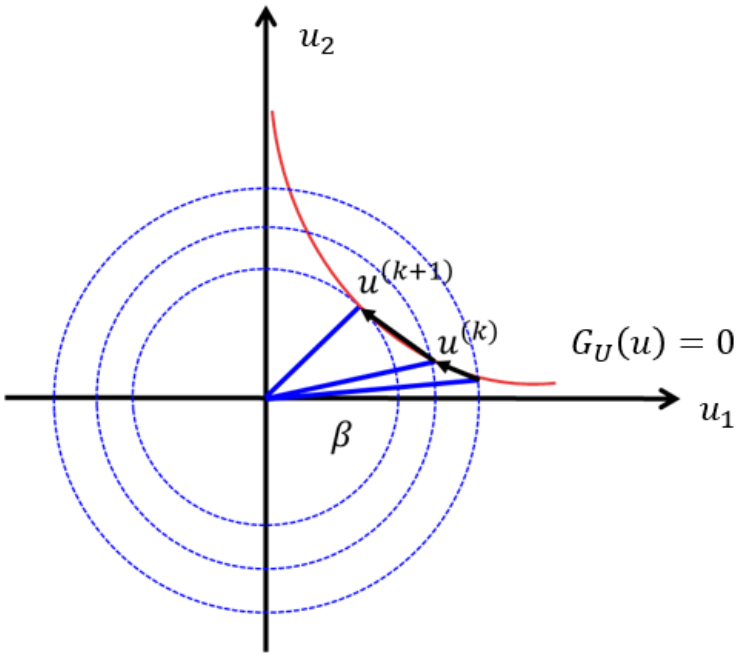


Fig. 1.14. Inner loop iteration of RIA

For the outer loop of RIA, any constrained non-linear algorithm like SQP or others can be chosen to minimize the objective function f with the index β computed with the FORM:

$$\begin{aligned} &\min_{\boldsymbol{d}} f(\boldsymbol{d}) \\ &s.t. \begin{cases} \boldsymbol{g}(\boldsymbol{d}) < 0 \\ \boldsymbol{\beta}(\boldsymbol{d}) \geq \beta_t \end{cases} \\ &\boldsymbol{d}^L + \beta_t\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t\boldsymbol{\sigma} \end{aligned} \tag{1.38}$$

where $\boldsymbol{g}(\boldsymbol{d}) < 0$ is used to restrict the search space because the definition in equation (1.35) is only valid if the origin is located in the security domain.

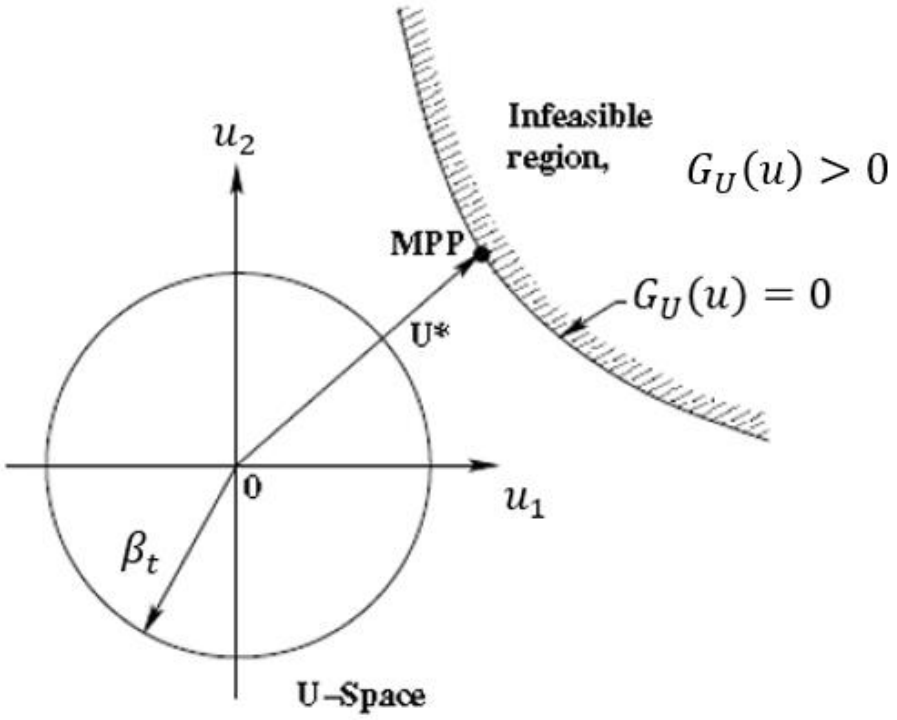


Fig. 1.15. RIA approach (Deb, Gupta, Dame, Branke & Mall, 2009)

In former formulation of RBDO, optimization is carried out with the limitation of the reliability index that must be greater than or equal to a targeted value. The calculation of this index leads to the search for the MPP. In contrast, for PMA formulation, optimization is formulated with the limitation of maximum performance with a given value of reliability index. The search of this maximum performance is to maximize the function G_U with the limitation that reliability index must achieve the targeted value. This approach is considered as the inverse of the FORM approximation (Choi & Youn, 2001; Wu, Millwater, & Cruse, 1990).

So the outer loop becomes:

$$\begin{gathered} \min_{d} f(\boldsymbol{d}) \\ s.t. \;\; G_p \leq 0 \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.39}$$

where G_p is the maximal performance measurement. The purpose of the inner loop is to find G_p in U-space by solving:

$$\begin{gathered} G_p = G_U(\boldsymbol{u}^*) = \max_{u} G_U(\boldsymbol{u}) \\ s.t. \;\; \|\boldsymbol{u}\| = \beta_t \end{gathered} \tag{1.40}$$

where $\boldsymbol{u}^*$ is the Maximum Performance Target Point (MPTP) that corresponds to the targeted index β_t. Also there are other methods besides the optimization algorithms to find the solution, the most popular one is the Advanced Mean Value (AMV) method (WU et al., 1990) and its derivatives.

These methods are also iterative approaches. The formulation of the first-order AMV method begins with the mean value point:

$$\boldsymbol{u}_{AMV}^{(0)} = 0, \boldsymbol{u}_{AMV}^{(1)} = \beta_t \boldsymbol{n}^{(0)} \tag{1.41}$$

where $\boldsymbol{n}^{(0)} = \frac{\nabla_X g(\boldsymbol{d})}{\|\nabla_X g(\boldsymbol{d})\|} = \frac{\nabla_U G_U(\boldsymbol{0})}{\|\nabla_U G_U(\boldsymbol{0})\|}$ is the normalized steepest ascent direction. The AMV method iteratively updates the ascent direction at the probable point $u_{AMV}^{(k)}$ starting from $\boldsymbol{u}_{AMV}^{(1)}$:

$$\boldsymbol{u}_{AMV}^{(k+1)} = \beta_t \boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k)}\right) \tag{1.42}$$

where $\boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k)}\right) = \frac{\nabla_U G_U\left(\boldsymbol{u}_{AMV}^{(k)}\right)}{\left\|\nabla_U G_U\left(\boldsymbol{u}_{AMV}^{(k)}\right)\right\|}$.

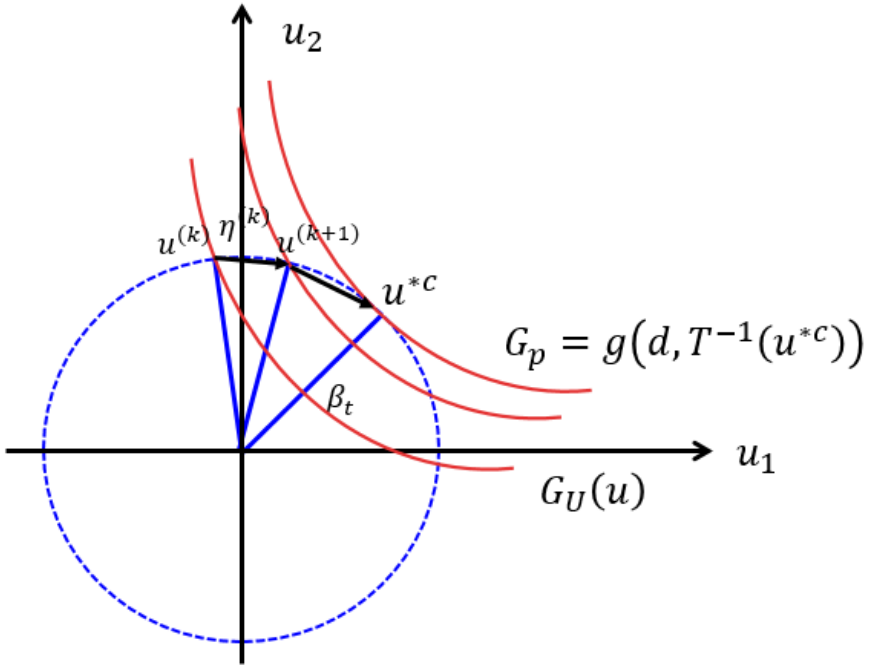


Fig. 1.16. Inner loop iteration of PMA by using AMV

However, as this method only uses the current MPTP to update the direction, it exhibits instability and when solving a concave performance function it tends to converge very slowly. In order to add updated information during the iterative optimization for accelerating the convergence, Conjugate Mean Value (CMV) method is proposed (Choi & Youn, 2001). It uses the combination of $\boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k)}\right), \boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k-1)}\right)$ and $\boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k-2)}\right)$ to calculate

the new search direction, so that it points to the diagonal of the three consecutive steepest ascent directions:

$$\begin{gathered}\boldsymbol{u}_{CMV}^{(0)}=\mathbf{0},\boldsymbol{u}_{CMV}^{(1)}=\boldsymbol{u}_{AMV}^{(1)},\boldsymbol{u}_{CMV}^{(2)}=\boldsymbol{u}_{AMV}^{(2)}\\ \boldsymbol{u}_{CMV}^{(k+1)}=\beta_t\frac{\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k)}\right)+\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k-1)}\right)+\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k-2)}\right)}{\left\|\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k)}\right)+\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k-1)}\right)+\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k-2)}\right)\right\|},for\ k\geq 2\end{gathered}\tag{1.43}$$

where $\boldsymbol{n}\left(\boldsymbol{u}_{CMV}^{(k)}\right)=\boldsymbol{n}\left(\boldsymbol{u}_{AMV}^{(k)}\right)$ has the same definition as in AMV.

Although CMV can significantly improve the rate of convergence compared to the AMV, it is found to be inefficient for the convex performance function. Thus, a method combining AMV and CMV is proposed to treat both the convex and concave performance functions. This method called Hybrid Mean Value (HMV) method adds a criterion to identify the type of performance function as follows (Choi & Youn, 2001; Youn, Choi, Yang, & Gu, 2004; Youn & Choi, 2004a):

$$\zeta^{(k+1)}=\left(\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k-1)}\right)-\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k)}\right)\right)\cdot\left(\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k)}\right)-\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k-1)}\right)\right)\tag{1.44}$$

This criterion employs steepest ascent directions at the three consecutive iterations and can determine whether the performance function is convex or concave at $\boldsymbol{u}_{HMV}^{(k)}$ as $\zeta^{(k+1)}$ represents the second order derivative. If $\text{sign}\left(\zeta^{(k+1)}\right)>0$, this means that the performance function is convex at $\boldsymbol{u}_{HMV}^{(k+1)}$, conversely it is concave.

The flow chart of this method is shown in the figure below.

First, the algorithm is initialized with $k=0$ and $\boldsymbol{u}_{HMV}^{(0)}=\mathbf{0}$, then it computes the steepest ascent direction of the performance function in U-space $\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k)}\right)$ as before. If the performance function is convex or $k<3$, HMV uses the same method as AMV to calculate the direction and $\boldsymbol{u}_{HMV}^{(k+1)}$. Then if the performance function is concave and $k\geq 3$, CMV method is used. After, $G_U\left(\boldsymbol{u}_{HMV}^{(k+1)}\right)$ at the new MPTP is calculated and a convergence criterion will be checked. If the criterion holds, then the algorithm stops, otherwise it sets $k=k+1$, computes the gradient $\nabla_U G_U\left(\boldsymbol{u}_{HMV}^{(k)}\right)$ and $\boldsymbol{n}\left(\boldsymbol{u}_{HMV}^{(k)}\right)$ for the next iteration.

The convergence criterion is:

$$\max\left(\left|\Delta G_{Urel}^{(k+1)}\right|,\left|\Delta G_{Uabs}^{(k+1)}\right|\right)\tag{1.45}$$

where

$$\begin{gathered}\Delta G_{Urel}^{(k+1)}=\frac{G_U\left(\boldsymbol{u}_{HMV}^{(k+1)}\right)-G_U\left(\boldsymbol{u}_{HMV}^{(k)}\right)}{G_U\left(\boldsymbol{u}_{HMV}^{(k+1)}\right)}\\ \Delta G_{Uabs}^{(k+1)}=G_U\left(\boldsymbol{u}_{HMV}^{(k+1)}\right)-G_U\left(\boldsymbol{u}_{HMV}^{(k)}\right)\end{gathered}\tag{1.46}$$

If the criterion is smaller than a given very small positive value ε, the convergence is considered achieved.

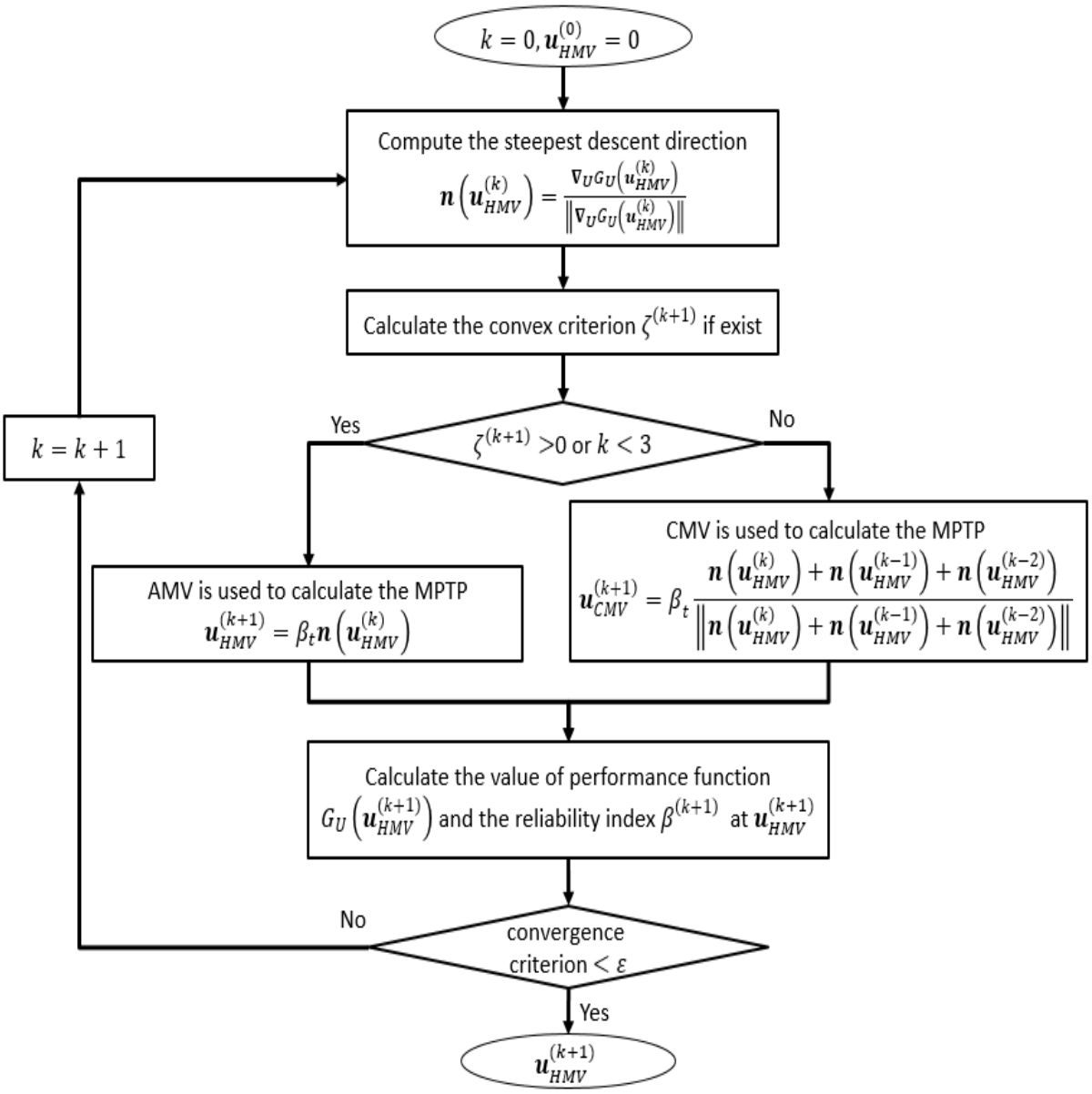


Fig. 1.17. The flowchart of HMV

Comparing the two double-loop methods, the reliability analysis in RIA may fail to converge with non-normal distributions. It is because that RIA must be inversely transformed into X-space to perform a design optimization, and this step introduces additional nonlinearity for all probability distributions except normal one (Youn & Choi, 2004a), as uniform distribution for example. PMA is more feasible as it does not involve non-linear implicit constraints.

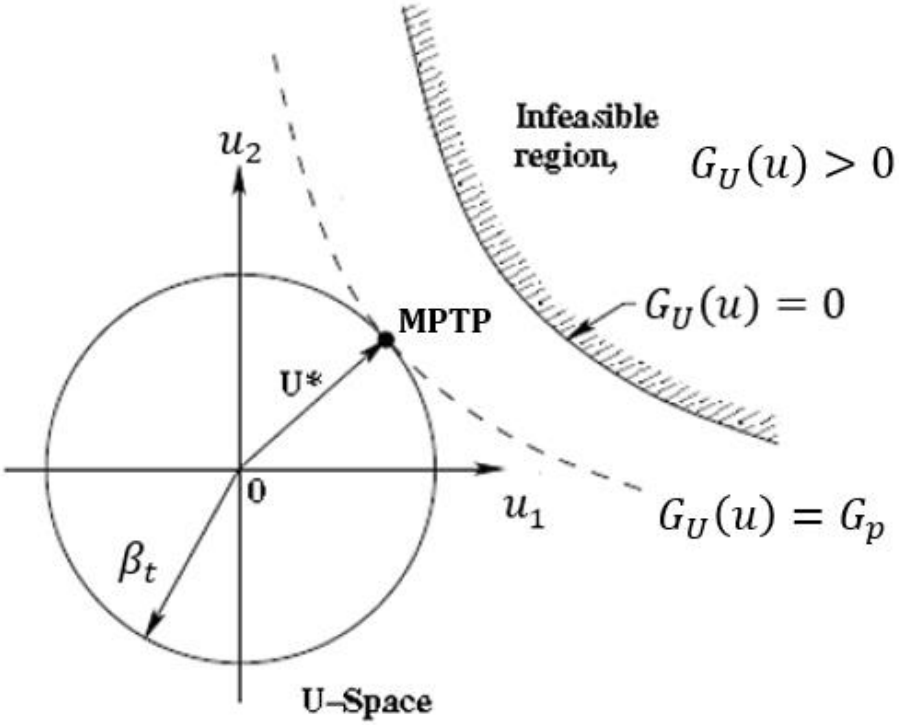


Fig. 1.18. PMA approach (K. Deb et al., 2009)

1.3.3 Single-loop method

For the so-called single-loop methods, the main point is that the inner loop is replaced by an approximation to avoid the iterative evaluations for reliability analysis in order to accelerate the convergence to the optimum.

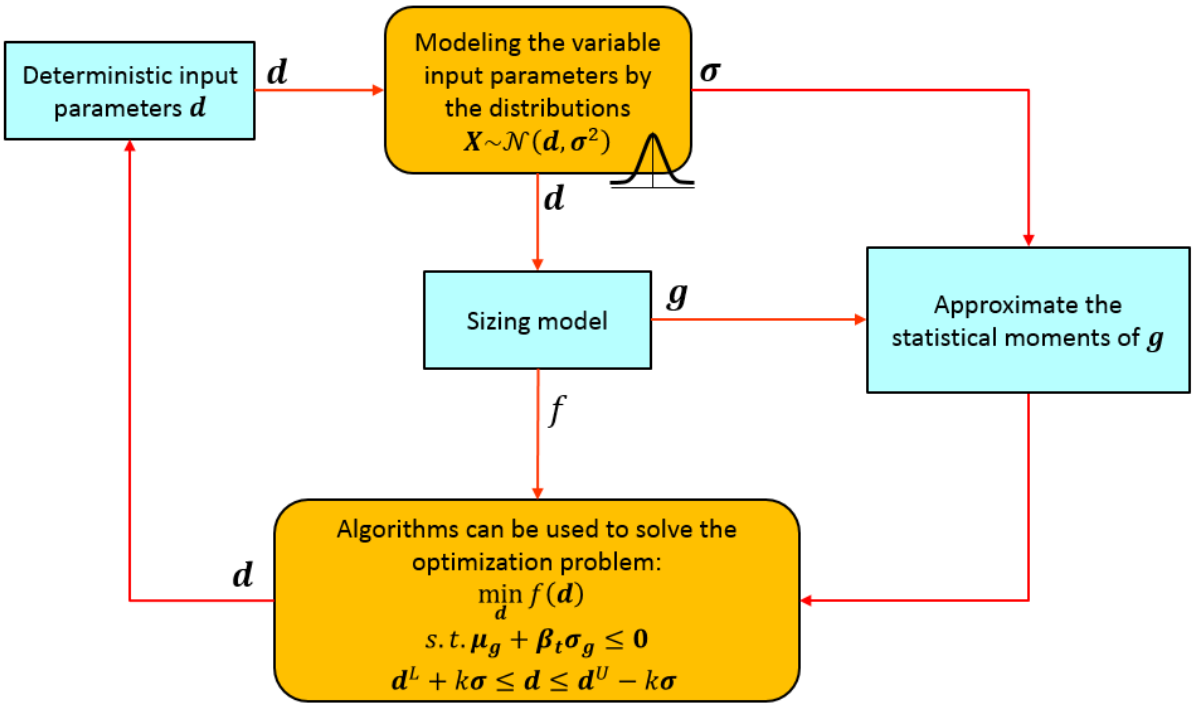


Fig. 1.19. The process of single-loop method

It is known that this type of methods focuses on reducing computing time rather than having a good precision. There are also several single-loop methods, some of them use the robust optimization methods often called the Six-Sigma approach to formulate the problem of RBDO in a single loop. These methods rely on the approximation of the statistical moments of the response, transforming the probabilistic constraints into purely deterministic functions. (Kuschel & Rackwitz, 1997) proposed to use Karush-Kuhn-Tucker optimality conditions of the first order reliability method (FORM) to replace the reliability constraints. (Chen et al., 1997) proposed the Single Loop Single Vector approach that uses derivatives of limit-state to calculate the reliability index. Based on the same concept, (Liang, Mourelatos, & Nikolaidis, 2007; Liang, Mourelatos, & Tu, 2004) proposed Single Loop Approach. (Shan & Wang, 2008) reformulate the RBDO problem by the concept of reliable design space. The RBDO is transformed to a simple deterministic problem constrained by the reliable design space rather than its deterministic feasible space. The most commonly used single-loop methods are Approximate Moments Approach (AMA) and Single Loop Approach (SLA). These two methods will be presented in details.

AMA is based on statistical moments. The first order Taylor expansion is used to calculate the mean μ_g and the standard deviation σ_g for all constraint functions $g(d)$ by using the following expressions (Putko, Taylor, Newman, & Green, 2002):

$$\begin{cases} \boldsymbol{\mu_g} = \boldsymbol{g}(\boldsymbol{d}) \\ \boldsymbol{\sigma_g^2} \approx \left(\boldsymbol{\nabla_g^T}\right)^2 \cdot \boldsymbol{\sigma}^2 \end{cases} \tag{1.47}$$

where σ is the standard deviation of the input variables X and ∇ is the gradient operator. As shown in Fig 1.20, with these expressions, a first order approximation $g(\boldsymbol{d}^*) + \beta_t\sigma_g$ is used to replace the value $g(\boldsymbol{d}^*)$. In this way, the limit-state of constraint is shifted in order to keep the reliability as desired. Fig. 1.20 shows the basic idea of AMA approach where $g(\boldsymbol{d}) = 0$ is the deterministic limit-state equation. As the optimum point $\boldsymbol{d}^*$ is on the limit-state curve, the probability of failure is about 50%. The situation which could cause the failure around this point with a given probability is shown as shaded area on the left side (Fig 1.20). AMA transforms the constraints in order to find a new optimum point $\boldsymbol{d'}^*$ on the curve of the new limit-state equation $g(\boldsymbol{d}) + \beta_t\sigma_g = 0$. The events which could violate the constraints around $\boldsymbol{d'}^*$ are greatly reduced as shown by the shaded area on the right side.

Therefore, the problem becomes a deterministic problem:

$$\begin{gathered} \min_{d} f(\boldsymbol{d}) \\ s.t.\ \ g(\boldsymbol{d}) + \beta_t\sigma_g \leq 0 \\ \boldsymbol{d}^L - \beta_t\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U + \beta_t\boldsymbol{\sigma} \end{gathered} \tag{1.48}$$

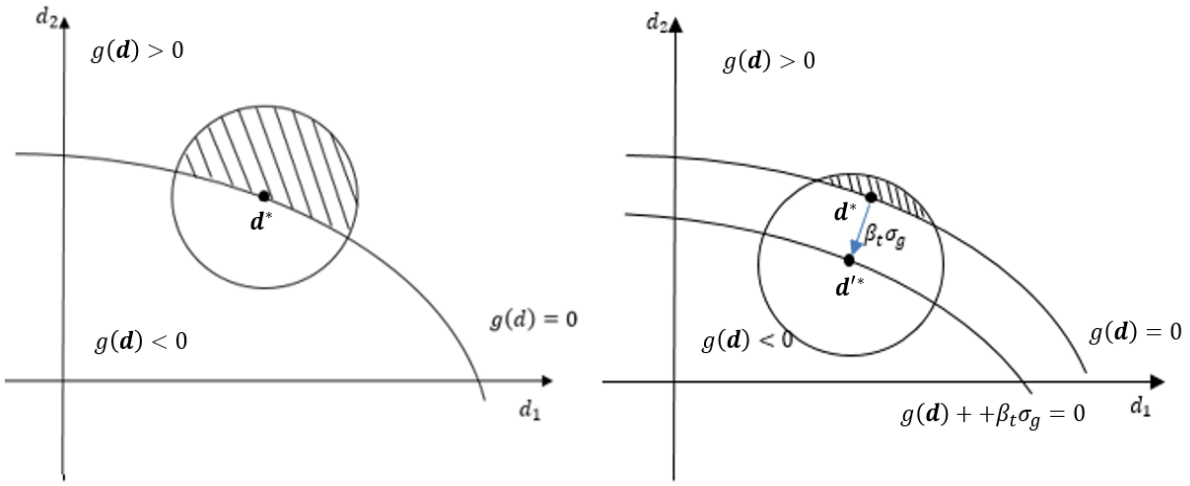


Fig. 1.20. Principle of the AMA approach

This approach is based on a local linear approximation of the constraint functions around the mean value of the design parameters. As the probabilistic distribution of the performances depends on two moments that are also approximated, the reliability assessment could produce significant numerical errors (Byeng D. Youn & Choi, 2004b).

SLA uses the same approximation as AMA for calculating the moments and the position of MPTP (marked as $\boldsymbol{d}'^*$ in Fig. 1.20 and Fig. 1.21). The difference between these two methods is that SLA evaluates the constraints g at the approximate MPTP unlike AMA that uses a first-order approximation of the constraint function around the mean value d instead. The SLA formulation is given as (Liang et al., 2004):

$$\begin{gathered} \min_{d} f(\boldsymbol{d}) \\ s.t.\ \ g_i(\boldsymbol{d_i}) \leq 0 \\ \boldsymbol{d}^L + \beta_t\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t\boldsymbol{\sigma} \end{gathered} \tag{1.49}$$

with

$$\boldsymbol{d_i} = \boldsymbol{d} + \beta_t\boldsymbol{\sigma} \circ \boldsymbol{n_i} \tag{1.50}$$

where $\boldsymbol{n_i}$ is the steepest ascent direction for the constraint g_i as defined before, $\boldsymbol{d_i}$ is the approximation of the MPTP for the constraint g_i, and $\circ$ is the Hadamard operator (element-wise) product.

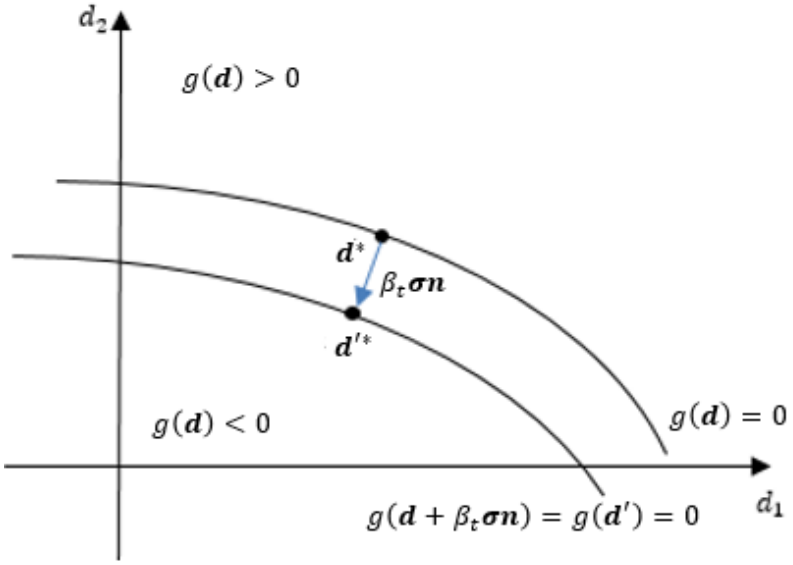


Fig. 1.21. Principle of the SLA approach

The principle of this approach is similar to AMA. As shown in Fig. 1.21, it does not search for the MPP of each constraint by using an inner loop but approximate its position. However, SLA needs more evaluations than AMA because the former evaluates the constraint at the approximated MPTP while AMA use another first order expansion to approximate this point. Thus, AMA uses two approximations whereas SLA uses only one.

It should be noticed that as all the methods of single-loop aim to accelerate the rate of convergence in reliability analysis by introducing approximations instead of searching the MPTP by solving an optimization problem. Therefore, the accuracy is sacrificed and the results given by these methods may not be precise enough especially when the design optimization problem is complex. The trade-off between the accuracy and the rate of convergence should be made by designers.

1.3.4 Sequential decoupled method

Sequential decoupled methods aim to change the initial nested optimizations into a series of optimization sequences.

(Wu & Wang, 1996) proposed the concept of Safety Factor Approach. It uses a series of deterministic optimization where the reliability constraints are substituted by equivalent deterministic constraints with safety factors that are calculated by one reliability analysis made before the beginning of the optimization loop. Based on the same idea of safety factor, (Qu & Haftka, 2004) proposed another method called the Probabilistic Sufficiency Factor to link the reliability requirement with deterministic optimization. This factor is calculated by efficient Monte Carlo simulations combined with response surface approximation. (Du & Chen, 2004) proposed the Sequential Optimization and Reliability Assessment (SORA) that is one of the most promising and commonly used sequential decoupled method. The cycles are sequential and independent. Each individual

optimization is deterministic and uses the optimum given by the former optimization as an initial point. At the first iteration, the algorithm searches a deterministic solution without considering uncertainty and then compute the maximum performance of this solution to deduce a shift. This shift lead the solution to move from the limit-states to security domain in order to achieve a given probability of failure. The next iterations refine the shift.

Another type of method proposed by (Cheng, Xu, & Jiang, 2006; Yi, Cheng, & Jiang, 2008) is Sequential Approximate Programming (SAP). The reliability constraints are replaced by first order Taylor expansions at the current point and the approximate reliability indexes can be obtained from the optimality conditions set at the MPP or MPTP. Then, the initial problem is reformulated as a series of approximate sub-problems where the optimal design is improved during each sub-problem. The two most popular approaches in this category, namely SORA and SAP, will be presented in details.

SORA employs a series of cycles of deterministic optimizations and reliability assessments. In each cycle, optimization and reliability assessment are decoupled from each other. The reliability assessment is only conducted after the deterministic optimization to verify constraint feasibility under uncertainty. The main point of this method is to shift the boundaries of constraints to the feasible direction based on the reliability information obtained in the former cycle (Du & Chen, 2004). The updated point is used in the next cycle of the deterministic optimization. This cycle is repeated until the fulfilment of the convergence criterion.

The process of SORA method is presented as follows. First, an initial shift $\boldsymbol{t}^0 = 0$ allows finding the solution $\boldsymbol{d}^0$ of the deterministic problem. At each iteration, the optimization problem is defined by:

$$\begin{gathered} \boldsymbol{d}^k = \underset{\boldsymbol{d}}{\operatorname{argmin}}\, f(\boldsymbol{d}) \\ s.t.\ \ g(\boldsymbol{d} - \boldsymbol{t}^k) \leq 0 \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.51}$$

The optimal value is set as $\boldsymbol{d}^k$. After each optimization, the optimal point $\boldsymbol{d}^k$ lies on some of the limit-state curves, it is called the constraint is active. When considering the randomness of X, the actual probability of failure of an active constraint is about 0.5. Then a reliability assessment is implemented at the deterministic optimum solution to locate the MPTP corresponding to the desired probability of failure giving $\boldsymbol{u}^*$ by solving (1.40). To ensure the MPTP is on the deterministic constraint boundary, a shifting vector $\boldsymbol{t}^{k+1}$ is derived:

$$\boldsymbol{t}^{k+1} = -\boldsymbol{u}^* \circ \boldsymbol{\sigma} \tag{1.52}$$

Therefore, when establishing the equivalent deterministic optimization model in the next cycle, the constraints is modified to ensure that the MPTP is on the deterministic constraint boundary.

A very important advantage of the SORA formulation is that the whole process is separated into several iterations of deterministic optimization and it can be easily implemented and solved by any classical optimization algorithm (Aoues & Chateauneuf, 2010).

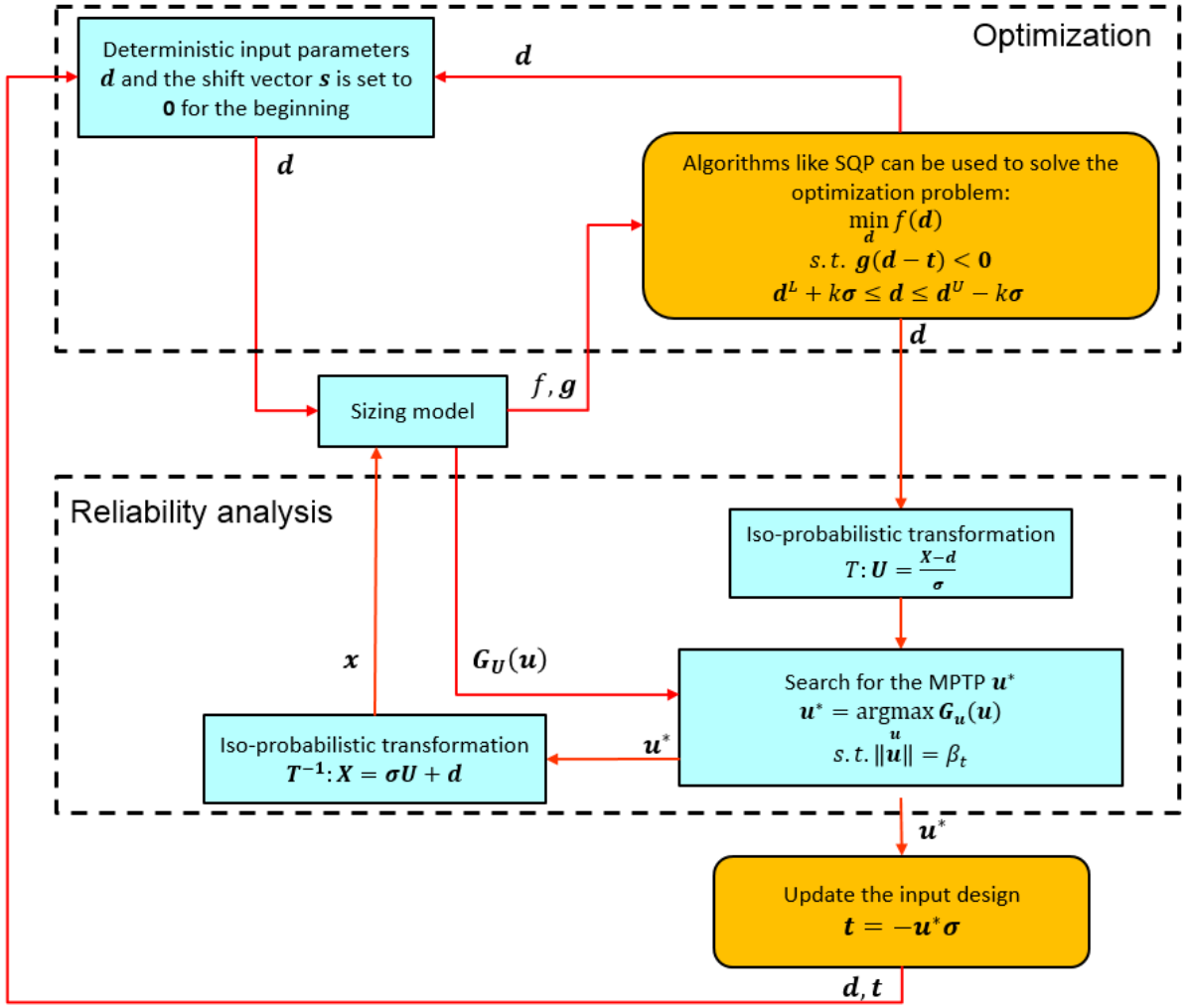


Fig. 1.22. The process of SORA

SAP is another approach based on Taylor expansion at the first order. The original optimization problem is decomposed into a sequence of sub-problems. Each sub-problem is carried out in the vicinity of the current design point $\boldsymbol{d}^k$ with an approximate objective function subjected to a set of approximate constraint functions (Yi et al., 2008). The details are presented as follows:

When considering the PMA optimization problem, the expressions are as in equation (1.39). Following the idea of SAP, a sequential approximate formulation is constructed as:

$$\begin{aligned} &\min_{d^k} f(\boldsymbol{d}^k) \\ s.t.\quad & G_p(\boldsymbol{d}^k) \le 0 \\ & \boldsymbol{d}^L + \beta_t\boldsymbol{\sigma} \le \boldsymbol{d}^k \le \boldsymbol{d}^U - \beta_t\boldsymbol{\sigma} \end{aligned} \tag{1.53}$$

However, if the original performance measure $G_p(\boldsymbol{d}^k) = G_U(\boldsymbol{u}^k)$ is used here, an inner optimization problem like (1.40) in PMA will be required. To avoid an inefficient bi-level algorithm, the performance measure is replaced here by a first order approximation:

$$\hat{G}_p(\boldsymbol{d}^k) = \hat{G}_p(\boldsymbol{d}^{k-1}) + \left(\boldsymbol{\nabla_d}\hat{G}_p(\boldsymbol{d}^{k-1})\right)^T (\boldsymbol{d}^k - \boldsymbol{d}^{k-1}) \tag{1.54}$$

where the approximation is obtained from the derivative $\boldsymbol{\nabla_d}\hat{G}_p(\boldsymbol{d}^{k-1})$, the approximated $\hat{G}_p(\boldsymbol{d}^{k-1})$ and optimum $\boldsymbol{d}^{k-1}$ of the $(k-1)th$ sub-problem, also the current design point $\boldsymbol{d}^k$ respectively.

The derivative is calculated by:

$$\frac{\partial \hat{G}_p(\boldsymbol{d}^{k-1})}{\partial \boldsymbol{d}^{k-1}} = \frac{\partial \hat{G}_U(\boldsymbol{u}^{k-1})}{\partial \boldsymbol{u}^{k-1}} \frac{\partial \boldsymbol{u}^{k-1}}{\partial \boldsymbol{d}^{k-1}} = \frac{\partial \hat{G}_U(\boldsymbol{u}^{k-1})}{\partial \boldsymbol{u}^{k-1}} \sigma \tag{1.55}$$

where $\boldsymbol{u}^{k-1}$ can be calculated by (1.42) in the same way as AMV method with $\boldsymbol{n}^k$ the steepest ascent direction of the performance function $G_U(\boldsymbol{u})$ at the point $\boldsymbol{u}^{k-1}$, and $\boldsymbol{u}^0 = 0$ is usually chosen as the initial estimation.

With the first order approximation of $G_p(\boldsymbol{d}^k)$, a new formulation is obtained:

$$\begin{gathered} \boldsymbol{d}^k = \underset{d}{\operatorname{argmin}}\, f(\boldsymbol{d}) \\ s.t.\ \ \hat{G}_p(\boldsymbol{d}^{k-1}) + \left(\nabla_d \hat{G}_p(\boldsymbol{d}^{k-1})\right)^T (\boldsymbol{d} - \boldsymbol{d}^{k-1}) \le 0 \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \le \boldsymbol{d} \le \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.56}$$

This formulation of SAP is similar to the outer loop of PMA. The same principle can be used to convert a RIA formulation into a sequential approximate programming (Cheng et al., 2006) with the following the sub-problems:

$$\begin{gathered} \boldsymbol{d}^k = \underset{d}{\operatorname{argmin}}\, f(\boldsymbol{d}) \\ s.t.\ \ \hat{\beta}(\boldsymbol{d}^{k-1}) + \left(\nabla_d \hat{\beta}(\boldsymbol{d}^{k-1})\right)^T (\boldsymbol{d}^k - \boldsymbol{d}^{k-1}) \ge \beta_t \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \le \boldsymbol{d} \le \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.57}$$

where

$$\begin{gathered} \hat{\beta}(\boldsymbol{d}^k) = \frac{1}{\|\nabla_U G_U(\boldsymbol{u}^k)\|} [G_U(\boldsymbol{u}^k) - (\boldsymbol{u}^k)^T \cdot \nabla_U G_U(\boldsymbol{u}^k)] \\ \nabla_d \hat{\beta}(\boldsymbol{d}^k) = \frac{\nabla_d g(\boldsymbol{d}^k)}{\|\nabla_U G_U(\boldsymbol{u}^k)\|} \end{gathered} \tag{1.58}$$

In our case, as all random variables follow the normal law, $\partial \boldsymbol{d}/\partial \boldsymbol{u} = \boldsymbol{\sigma}$. Thus $\nabla_d \hat{\beta}(\boldsymbol{d}^k)$ can be written as:

$$\nabla_d \hat{\beta}(\boldsymbol{d}^k) = \frac{\nabla_U G_U(\boldsymbol{u}^k)}{\|\nabla_U G_U(\boldsymbol{u}^k)\|} \cdot \frac{1}{\boldsymbol{\sigma}} \tag{1.59}$$

1.4 Reliability-Based Robust Design Optimization (RBRDO)

As can be seen from the name, RBRDO is a mix of RDO and RBDO. It should be pointed out that the aims of RDO and RBDO are different.

"Robust" means that a function or a device is insensitive to the input variability. Like in the Fig 1.23, the PDF which is narrower is considered more robust as the variance of the output function is smaller. So robust design is a method to improve the quality of products by minimizing the effect from the variability of the input parameters. It emphasizes on enhancing the robustness of performance.

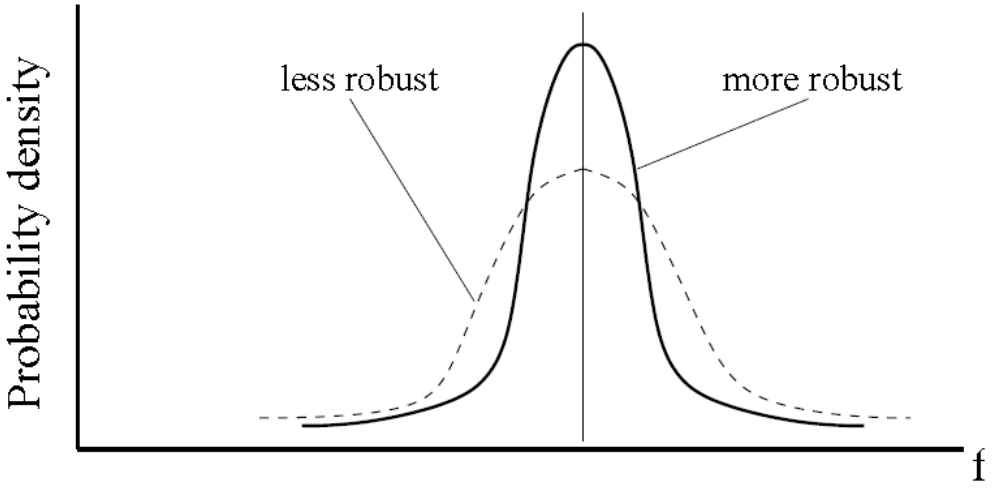


Fig. 1.23. The propose of RDO and WCO is to have a narrower PDF (Kang, 2005)

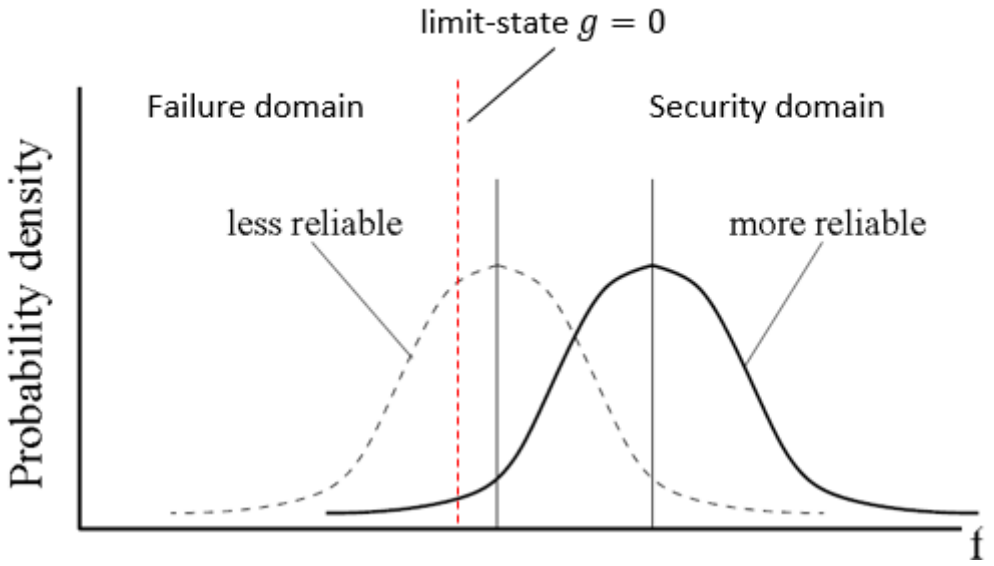


Fig. 1.24. The propose of RBDO is to keep away from the limit-states (Kang, 2005)

On the other hand, "reliable" means that it has a small probability of failure. As shown in Fig 1.24, the red dotted line is the limit-state $g = 0$ and the left part is the failure domain and on the contrary the right part is the security domain. The left PDF is less reliable because the probability to violate the constraints is higher than the right one. So the reliability-based design approach focuses on maintaining design feasibility at an expected probabilistic level (Du, Sudjianto, & Chen, 2004).

It is desirable to integrate the two methods together in order to have a more robust objective and reliable constraints.

So it is obvious that the combination of these two approaches ensure both reliability and robustness during the design optimization (Yadav, Bhamare, & Rathore, 2010) and avoid their own shortcomings. Essentially, the type of objective function used in RBDO is replaced by the type used in RDO in the hybrid formulation (Enevoldsen, 1994; Tu et al., 1999; Yu, Chang, & Choi, 1998):

$$\begin{gathered} \min_{d} f_r(\mu_f, \sigma_f) \\ s.t. \quad P_f \leq P_t \\ \boldsymbol{d}^L + \beta_t \boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - \beta_t \boldsymbol{\sigma} \end{gathered} \tag{1.60}$$

Since the design optimization aims to manufacture the product that satisfies customer's needs and expectations then translate them into appropriate quality characteristics. It is important to understand and classify the nature of these product quality characteristics that need to be optimized. The classification helps to understand and capture the desirability of variables unlike the traditional formulations that seek to minimize the total deviation. Normally the types of classification include nominal-the best, larger-the-better, and smaller-the-better quality characteristics. Based on this classification, authors proposed three type of cost function that will be detailed in the following (Chandra, 2001; Byeng D. Youn, Choi, & Yi, 2005).

1.4.1 Nominal-the-best type

For nominal-the-best (NTB) quality characteristics, the aim is to achieve the targeted value f_t for the objective function value. This requirement classifies that both positive and negative deviations are considered as undesirable, here the deviation is defined as the difference between the targeted value of any given objective and the value obtained by the model solution (Bhamare, Yadav, & Rathore, 2009). A commonly used formulation for this type is written as:

$$f_r(\mu_f, \sigma_f) = \omega_1 \left(\frac{\mu_f - f_t}{\mu_{f_0} - f_t}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2 \tag{1.61}$$

where f_t is the targeted value of the cost function, ω_1 and ω_2 are the weights chosen by the designer, μ_{f_0} and σ_{f_0} are the scaling values which can be chosen equal to the mean and standard deviation values at initial point, like Equation (1.28) in RDO.

1.4.2 Smaller-the-better type

For small-the–better (STB) type characteristic, the purpose is searching a smaller mean of objective. As shown in the Fig 1.25, μ_f^* is considered as a better solution than μ_{f_0}. Thus a new robust objective function is proposed in order to help us finding a smaller value:

$$f_r(\mu_f, \sigma_f) = \omega_1 \cdot \text{sign}(\mu_f) \left(\frac{\mu_f}{\mu_{f_0}}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2 \quad (1.62)$$

The objective is normalized again and for the negative response, the sign function of μ_f is multiplied in order to properly minimize the objective (Byeng D. Youn et al., 2005).

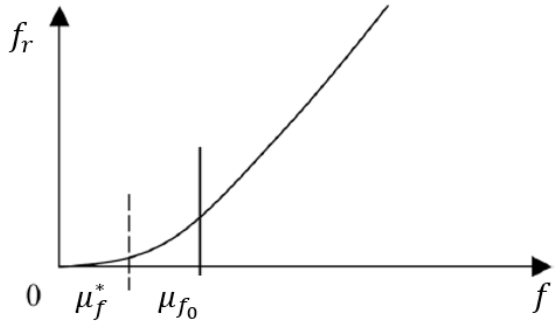


Fig. 1.25. Robust objective function for a smaller-the-better type characteristic (Bhamare et al., 2009)

1.4.3 Larger-the-better type

Larger-the-better (LTB) type is on the contrary of small-the-better one, the larger mean is more satisfying for designer. A new robust objective function shown in Fig 1.26 should be used so that minimizing f_r could lead to a larger value of mean. The objective function is (I. Lee, Choi, Du, & Gorsich, 2008):

$$f_r(\mu_f, \sigma_f) = \omega_1 \cdot \text{sign}(\mu_f) \left(\frac{\mu_{f_0}}{\mu_f}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2 \quad (1.63)$$

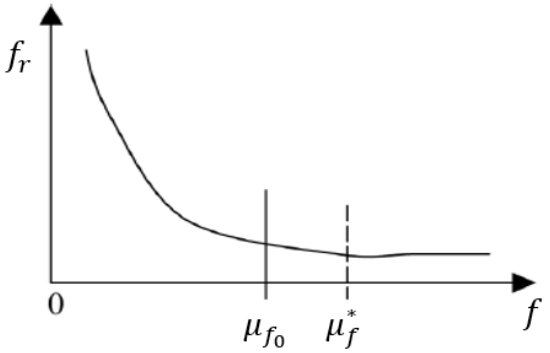


Fig. 1.26. Robust objective function for a larger-the-better type characteristic (Bhamare et al., 2009)

It is needed to remind that for the variance the aim is always minimization and no matter the larger-the-better, smaller-the-better or normal-the-best, they only refer to the mean. So that in the expressions just the first part with mean change.

The designer could choose the most suitable type of robust objective function in accordance with the real requirements. For example, if our goal is to minimize the volume of a device, the quality characteristic is considered as smaller-the-better so that the second type of formulation is the most suitable. Besides the mentioned types, the robust objective function $f_r(\mu_f, \sigma_f)$ can be formulated in various ways based on engineering application types. For example (Ziyan Ren, Zhang, & Koh, 2013) propose a new RBRDO algorithms for electromagnetic devices using Worst-Case Scenario Approximation. (Shahraki, Noorossana, 2013) decides to use a piecewise-linear function as the cost function. Also there are other formulations for the types of Normal-the-best and Larger-the-better like (Paiva, 2010) who proposes to use $\mu_f + \sigma_f$ directly as the objective function.

2 Numerical investigations on methods

In this section, a simple mathematic example with two variables from (Dong-Wook Kim, Nak-Sun Choi, Choi, & Dong-Hun Kim, 2015) is used to compare different formulations from each type of optimization method (WCO, RDO, RBDO, RBRDO). In order to understand the implications of the choice of different formulations and also to quantify the impact of the choice of approaches and algorithms on the accuracy and the number of evaluations, the results of the different type of optimization will be given separately.

The mathematic example has one objective function and three constraints:

$$\begin{gathered} f(\boldsymbol{d}) = -\frac{(d_1+d_2-10)^2}{30} - \frac{(d_1-d_2+10)^2}{120} \\ s.t.\ \boldsymbol{g}(\boldsymbol{d}) \le 0 \\ 0 \le d_{1,2} \le 10 \end{gathered} \tag{1.64}$$

with

$$\boldsymbol{g}(\boldsymbol{d}) = \begin{bmatrix} g_1(\boldsymbol{d}) \\ g_2(\boldsymbol{d}) \\ g_3(\boldsymbol{d}) \end{bmatrix} = \begin{bmatrix} 1 - \frac{d_1^2 d_2}{5} \\ 1 - \frac{(d_1+d_2-5)^2}{30} - \frac{(d_1-d_2-12)^2}{120} \\ 1 - \frac{80}{(d_1^2+8d_2+5)} \end{bmatrix} \tag{1.65}$$

where d_1 and d_2 are two continuous input variables, as shown in Fig 1.27, the failure domain is the shaded area, the counterpart is the security domain, the colour contours are contours of objective function f.

For considering the uncertainties, the two variables are supposed to be random variables that follow the normal law with the unknown mean $\boldsymbol{d}$ which we are looking for and the standard deviation $\boldsymbol{\sigma} = [0.3; 0.3]$.

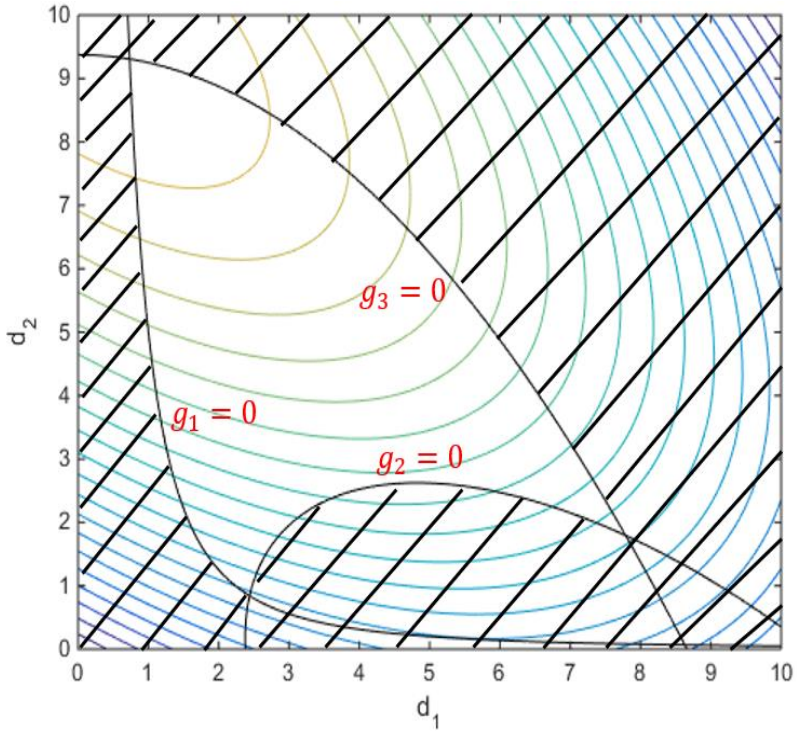


Fig. 1.27. Security domain of the mathematic example (shaded)

2.1 Comparison of WCO methods

Traditional WCO method, WWCO, and first-order GWCO presented in section 1.1 in this chapter are tested here. The confidence level k is equal to 2. Optimal solutions and the number of evaluations are given in the following table to compare the efficiency of methods.

As it can be seen in Table 2.2, traditional WCO needs the largest number of evaluations owing to the way it works. Indeed, the computation of constraints and objective is made within an inner-loop optimization while the search for the optimal value of d is an outer-loop optimization.

Another information we can get from the table is that the accuracy of WWCO is higher than GWCO. There are two columns of $f_w(\boldsymbol{d}_{opt})$ and $g_{w,i}(\boldsymbol{d}_{opt})$ where $i = 1,2,3$, the left one are the results calculated by GWCO itself and the right one is calculated by the traditional WCO. The difference of these two columns explains the phenomenon why GWCO is less accurate, it computes a forward gradient by using only $p + 1$ evaluations and overestimates the worst-case while WWCO uses $2p + m + 1$ evaluations for a better assessment where m is the number of constraints.

Table 1.2. Results of different formulations of WCO with fast model

Formulations	WCO	WWCO	GWCO	
$\boldsymbol{d}_{opt}$	[2.2664; 2.4006]	[2.2664; 2.4006]	[2.3524; 2.1568]	
$f(\boldsymbol{d}_{opt})$	-1.7592	-1.7592	-1.8712	
$f_w(\boldsymbol{d}_{opt})$	-1.3805	-1.3805	-1.1002	
$g_1(\boldsymbol{d}_{opt})$	-1.4661	-1.4661	-1.3869	
$g_2(\boldsymbol{d}_{opt})$	-0.2307	-0.2307	-0.1692	
$g_3(\boldsymbol{d}_{opt})$	-1.7265	-1.7265	-1.1966	-1.8790
$g_{w,1}(\boldsymbol{d}_{opt})$	0	0	0	-0.6905
$g_{w,2}(\boldsymbol{d}_{opt})$	0	0	0	-0.0136
$g_{w,3}(\boldsymbol{d}_{opt})$	-1.1493	-1.1493	-0.8010	-0.9017
evaluations	12951	231	205	

Fig 1.28 to 1.30 show the disparate way of approximation of worst-case constraints by two level optimization WCO (Fig 1.28), WWCO (Fig 1.29) and first-order GWCO (Fig 1.30). The red point is the optimum, the rectangle denotes the uncertainty set around the optimum. Dotted red lines are the real constraints and the solid blue lines present the approximation of the worst-case constraints.

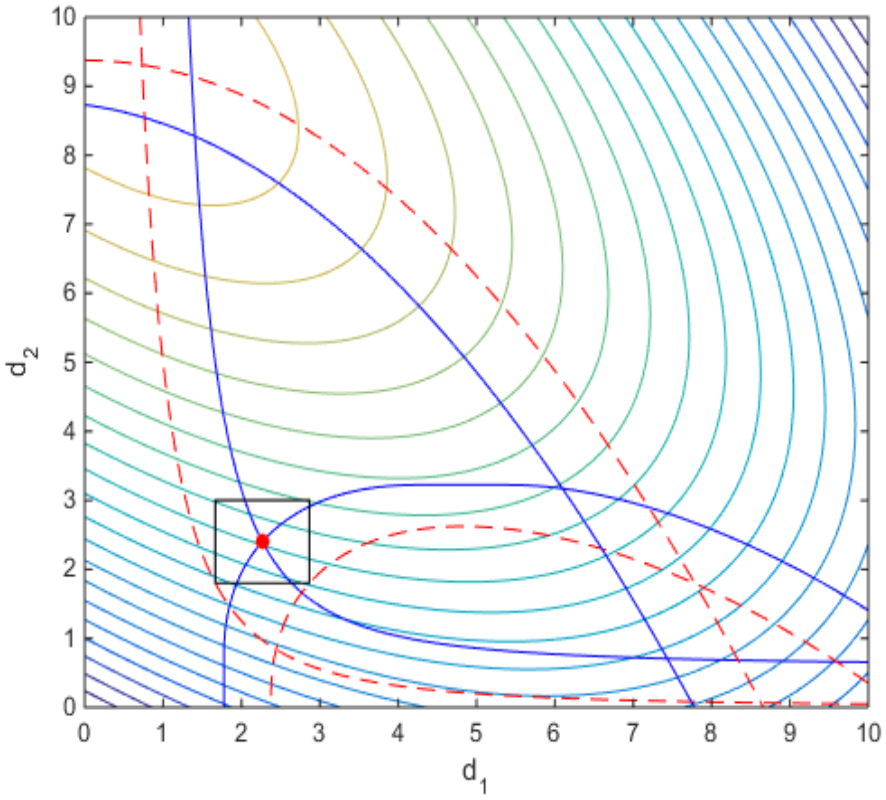


Fig. 1.28. Optimum and worst-case of constraints obtained by two level optimization WCO

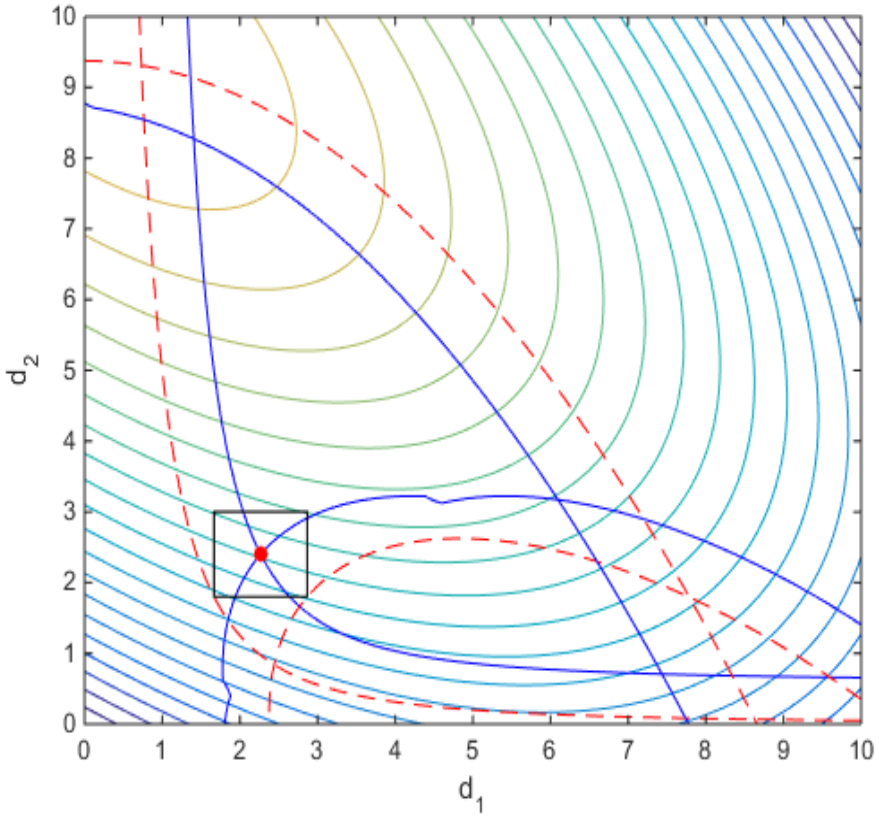


Fig. 1.29. Optimum and worst-case of constraints obtained by WWCO

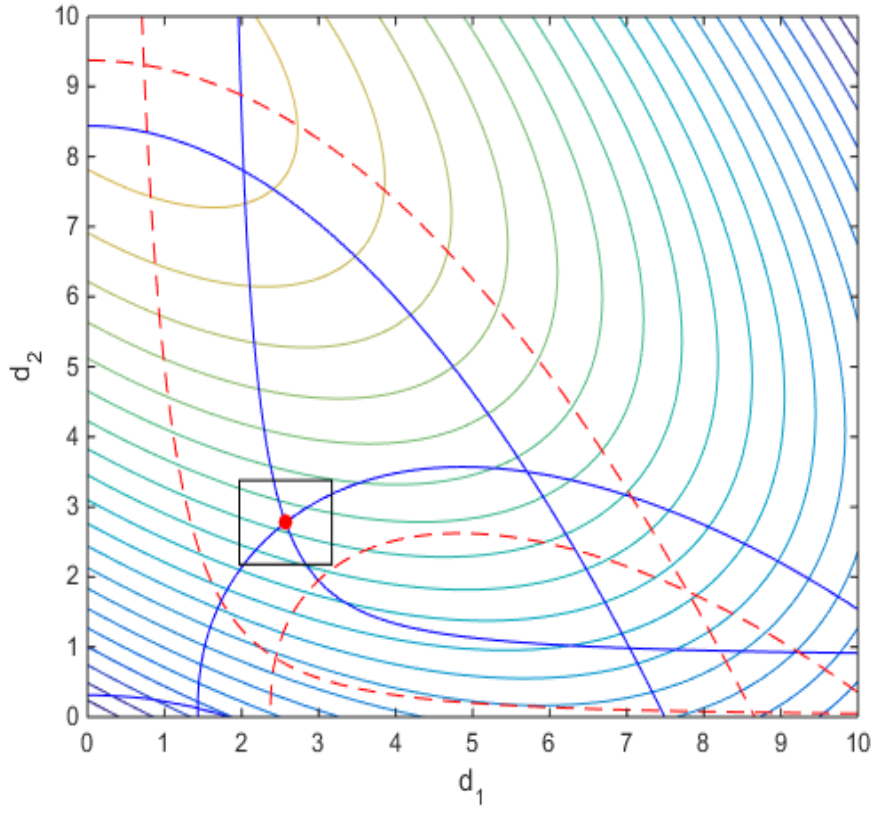


Fig. 1.30. Optimum and worst-case of constraints obtained by first order GWCO

The optima found by WCO and WWCO are the same and corners of the uncertainty set of optimum just lie on the limit-states. That means all of the possible cases in the uncertainty set can satisfy the constraints. However, from Table 2.2 and Fig 1.30 we could find that the results of GWCO are different from others and the corners of uncertainty set around optimum do not lie on the limit-states. That is because after using (1.11), the worst-case of constraints is:

$$\boldsymbol{g}_w(\boldsymbol{d}) = \boldsymbol{g}(\boldsymbol{d}) + \|\nabla \boldsymbol{g}(\boldsymbol{d})\| \cdot \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \|\boldsymbol{\xi} - \boldsymbol{d}\| \tag{1.66}$$

But the first order Taylor expansion at $\boldsymbol{g}(\boldsymbol{d} + k\boldsymbol{\sigma})$ is:

$$\boldsymbol{g}(\boldsymbol{d} + k\boldsymbol{\sigma}) = \boldsymbol{g}(\boldsymbol{d}) + \nabla \boldsymbol{g}(\boldsymbol{d}) \cdot k\boldsymbol{\sigma} \tag{1.67}$$

The second term on the right side of (1.60) is always greater or equal to the second term on the right side of (1.61), thus the distance of worst-case with design point calculated by GWCO is always larger than the distance $k\boldsymbol{\sigma}$.

From Fig 1.29 it can also be noticed that the limit-state $g_2 = 0$ is not smooth in all places. This is due to the method used to calculate the worst-case. The WWCO method only evaluate the bounds of each variable and then deducts roughly that the worst-vertex is the worst-case of all uncertainty set. However in some cases the worst-case may be a point in the set not the vertex, in fact if we look at the corresponding non-smooth area in Fig 1.28, it can be seen that the curvature is different with other areas. That means the position of worst-case is changing with the displacement of uncertainty set, but this changing process is ignored by WWCO and it sets all worst-cases as vertex as usual thus resulting in the non-smooth blue curve in Fig. 1.29.

2.2 Comparison of RDO methods

As there are various formulations for RDO, we choose only several representative ones from them. The synthesis of these formulations is shown in the Table 1.3. For some of them, mean and standard deviation of initial objective functions are scaled in order to reduce the difference of magnitude. The needed scaling values μ_{f_0} and σ_{f_0} can be calculated in the beginning by propagating the uncertainties of the chosen starting point.

The robust objective function of Formulation 1 considers only the mean of initial objective (Sundaresan, Ishii & Houser, 1992). Formulation 2 uses the same approximation of constraints and apply it for the objective function (Wang, Huang & Liu, 2010). Formulation 3 has the similar structure and the only difference is that we add scaled values in order to reduce the impact of magnitude difference between the mean and standard deviation. All these aforementioned formulations are mono-objective and do not introduce any other parameters, thus the results do not change with the designer's requirements.

Other formulations can lead to different results which depend on whether the designer focus on minimizing the mean or minimizing the standard deviation. Formulation 4 has the same robust objective functions than formulation 1, meanwhile it adds an extra constraint $\sigma_f(\boldsymbol{d}) - \sigma_t \leq 0$ which is called ϵ-constraint (Barba, 2009; Collette & Siarry, 2011). This method transforms the initial bi-objective optimization problem into series of single-objective problem with an additional constraint so that it could be handled by a single-objective optimization algorithm. For our example, the ϵ-constraint method keeps μ_f as objective while passing σ_f in constraint. The additional constraint function is imposed at different limit values from the maximum σ_{fmax} to the minimum σ_{fmin} of standard deviation. So that the optimum may changes with the modification of σ_t. The values σ_{fmax} and σ_{fmin} used in robust constraints can be calculated by maximizing and minimizing only the standard deviation of initial objective function before RDO begins. Formulation 5 proposed by (Sun et al., 2011) adds a weight factor $\omega \in [0,1]$ in order to balance the weight of mean and standard deviation. However, the authors do not consider the situation when $\mu_f(\boldsymbol{d})$ is negative and use only the square what may lead to a maximization if $\mu_f(\boldsymbol{d})$ is negative. Therefore, we change the formulation by adding the sign of $\mu_f(\boldsymbol{d})$. Formulation 6 and 7 also use the weight factor, but the former one uses the mean and standard deviation values while the latter uses their scaled values. The last two formulations consider both objectives separately, one with the original values and the other with scaled values.

Table 1.3. Different formulations used for RDO

Formulation	Robust objective functions	Robust constraints
1	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
2	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d}) + \beta_t \sigma_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
3	$f_r(\boldsymbol{d}) = \frac{\mu_f(\boldsymbol{d})}{\lvert\mu_{f_0}\rvert} + \beta_t \frac{\sigma_f(\boldsymbol{d})}{\sigma_{f_0}}$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
4	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d})$	$\boldsymbol{g_{r1}}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$ $\boldsymbol{g_{r2}}(\boldsymbol{d}) = \sigma_f(\boldsymbol{d}) - \sigma_t,$ $\sigma_t \in [\sigma_{fmin}, \sigma_{fmax}]$
5	$f_r(\boldsymbol{d}) = \omega \cdot sign\left(\mu_f(\boldsymbol{d})\right) \mu_f(\boldsymbol{d})^2 + (1-\omega)\sigma_f(\boldsymbol{d})^2$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
6	$f_r(\boldsymbol{d}) = \omega\mu_f(\boldsymbol{d}) + (1-\omega)\sigma_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
7	$f_r(\boldsymbol{d}) = \omega \frac{\mu_f(\boldsymbol{d})}{\lvert\mu_{f_0}\rvert} + (1-\omega) \frac{\sigma_f(\boldsymbol{d})}{\sigma_{f_0}}$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
8	$f_{r1}(\boldsymbol{d}) = \mu_f(\boldsymbol{d}), f_{r2}(\boldsymbol{d}) = \sigma_f(\boldsymbol{d})$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$
9	$f_{r1}(\boldsymbol{d}) = \frac{\mu_f(\boldsymbol{d})}{\lvert\mu_{f_0}\rvert}, f_{r2}(\boldsymbol{d}) = \frac{\sigma_f(\boldsymbol{d})}{\sigma_{f_0}}$	$\boldsymbol{g_r}(\boldsymbol{d}) = \boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d})$

The value of β_t is chosen equal to 2. The first two statistical moments, the mean and the standard deviation, are calculated by first-order Taylor based method. The results for each formulation will be presented sequentially.

The results of the first three formulations are given in Table 1.4 and shown in Fig 1.31. As there is no weight or other parameters in robust objective function or constraints, there is only one result for each formulation. All the three formulations use Sequential quadratic programming (SQP) algorithm, as it is a local optimization algorithm, 100 different uniformly sampled initial points are used here in order to converge to a global optimum, and the rates of convergence to the optimum with 100 runs are also shown in the table, it explains how many time it will converge to the same minimum.

Table 1.4. Results for mono-objective formulations of RDO

Formulation	$\boldsymbol{d_{opt}}$	$f_r(\boldsymbol{d_{opt}})$	$\mu_f(\boldsymbol{d_{opt}})$	$\sigma_f(\boldsymbol{d_{opt}})$	Evaluations	Convergence (with 100 runs)
1	[2.3524; 2.1568]	-1.8712	-1.8712	0.1712	1003	34
2	[2.3524; 2.1568]	-1.5288	-1.8712	0.1712	952	36
3	[2.3316; 2.0358]	0.4053	-1.4829	0.0955	1598	64

It should be noticed that the optima calculated by these formulations are not similar, Formulation 1 and 2 could find the same optimum and the moments calculated $\mu_f(\boldsymbol{d_{opt}})$ and $\sigma_f(\boldsymbol{d_{opt}})$ show that the optimization focus on the mean without considering the standard. It is easy to understand because Formulation 1 only focus on the mean and the two terms in Formulation 2 has different order of magnitude, absolute value of mean is much greater than standard deviation, so these two formulations get to the point which minimize the mean. Formulation 3 leads to a solution that presents a trade-off between the mean value and standard deviation thank to the scaling. As β_t is fixed, a single solution is found.

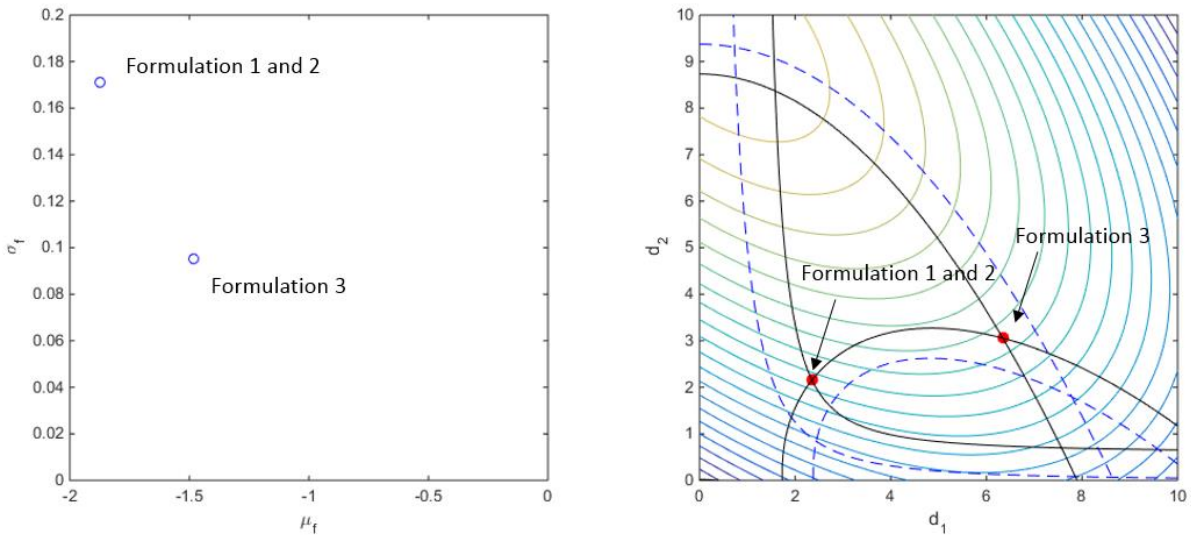


Fig. 1.31. Results for the formulation 1, 2 and 3

In the Fig. 1.31, the left part presents the mean $\mu_f(\boldsymbol{d_{opt}})$ and standard deviation $\sigma_f(\boldsymbol{d_{opt}})$ calculated at the optimum found. In the right part, the blue dotted lines are initial limit-states $g = 0$, and black solid lines are the robust limit-state $\boldsymbol{\mu_g}(\boldsymbol{d}) + \beta_t \boldsymbol{\sigma_g}(\boldsymbol{d}) = \boldsymbol{0}$, solutions found by different formulations are presented by the red points.

For Formulation 4， we first minimize and maximize the standard deviation of initial objective function to obtain σ_{fmin} and σ_{fmax} and divide the interval $[\sigma_{fmin}, \sigma_{fmax}]$ into 100 sections. σ_t equals to every bounds of these sections and the results of these 101 optimizations are shown in the figure below. As before, SQP algorithm is used and to be sure that the solutions is global, 20 runs with different initial points are used.

The left part of the Fig. 1.32 presents the mean and standard deviation of initial objective function calculated at optima found, the right part presents locations of optima. The optima distribute almost on the two robust limit-states g_{r2} and g_{r3}. However, the Pareto front obtained is not continuous because the points on the right part of g_{r2} are dominated by other solutions which can be seen from the Fig 1.32 (left). Some solutions lie on a line and not exactly on the limit-state g_{r3} because the constraint is not active for some minima of the robust objective function.

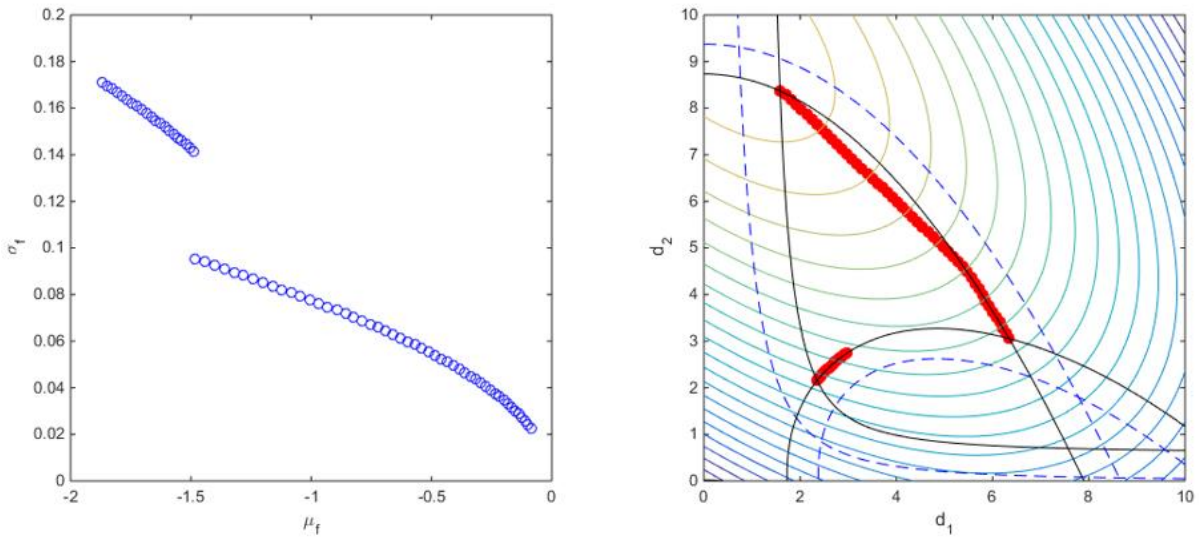


Fig. 1.32. Results for Formulation 4

To verify this view, we can modify the third constraint in Equation (1.65) into equality constraint as follows:

$$f(\boldsymbol{d}) = -\frac{(d_1+d_2-10)^2}{30} - \frac{(d_1-d_2+10)^2}{120}$$

$$s.t. \begin{cases} g_1(\boldsymbol{d}) = 1 - \frac{d_1^2 d_2}{5} \leq 0 \\ g_2(\boldsymbol{d}) = 1 - \frac{(d_1+d_2-5)^2}{30} - \frac{(d_1-d_2-12)^2}{120} \leq 0 \\ h(\boldsymbol{d}) = 1 - \frac{80}{(d_1^2+8d_2+5)} = 0 \end{cases} \quad (1.68)$$

$$0 \leq d_{1,2} \leq 10$$

This way, all solutions found should be on the robust equality constraint. In Fig 1.33, left part, the blue points are calculated with an equality constraint while the green points are calculated by initial problem. For the right part, red points are optima found by the modified problem with equality constraint and yellow points are optima found by the initial problem. It can be seen that optima found by the modified problem with equality constraint are exactly on the limit-state but the front Pareto with blue points is a little upper than the other one with green points, thus they are dominated so we can say that the optima found in Fig 1.32 are better than the solutions on the limit-state.

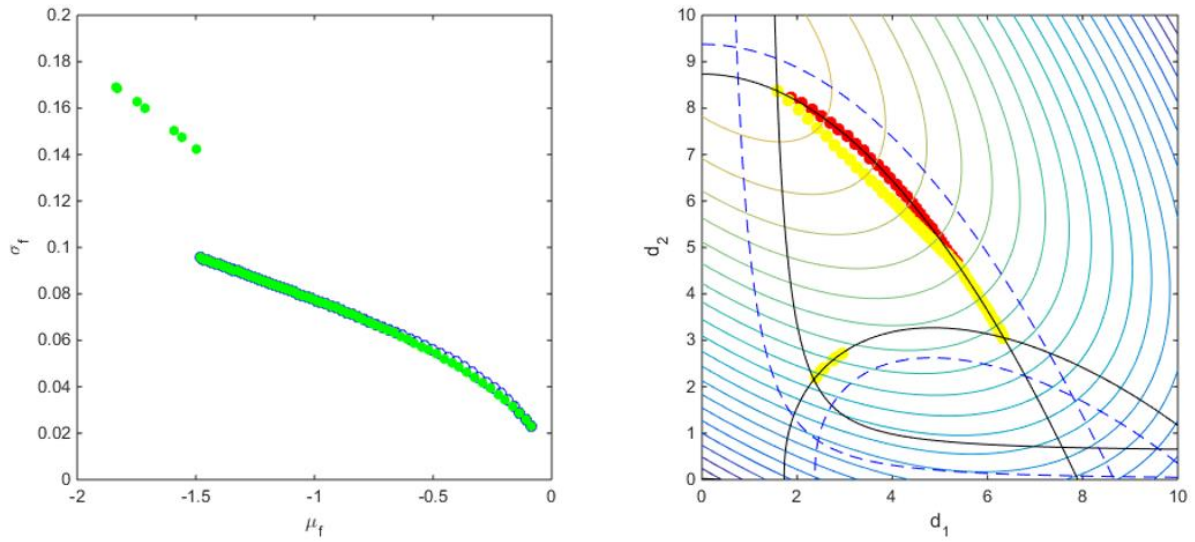


Fig. 1.33. Comparison of the results for Formulation 4

For Formulation 5, 6 and 7, the weight ω is chosen from 0 to 1 with a step of 0.01. Thus, 101 optimizations are also computed. Again, SQP algorithm is used with 20 runs from different initial points. The results of these three formulations are presented in Table 1.5 and the Fig. 1.34 to 1.36.

All these three formulations have the same solutions: three intersection points of the three robust constraints. In fact, these points are also the bounds of the front Pareto shown in Fig 1.32. The reason why the weight factor ω does not work is due to the concave shape of the front Pareto.

Table 1.5. Different values of weight for solutions

Formulation	$\boldsymbol{d_{opt}} = [1.5828; 8.3685]$	$\boldsymbol{d_{opt}} = [6.3417; 3.0564]$	$\boldsymbol{d_{opt}} = [2.3524; 2.1568]$
5	$\omega = 0$	$\omega = 0.01$	$\omega \in [0.02,1]$
6	$\omega \in [0,0.05]$	$\omega \in [0.06,0.16]$	$\omega \in [0.17,1]$
7	$\omega \in [0,0.32]$	$\omega \in [0.33,0.64]$	$\omega \in [0.65,1]$

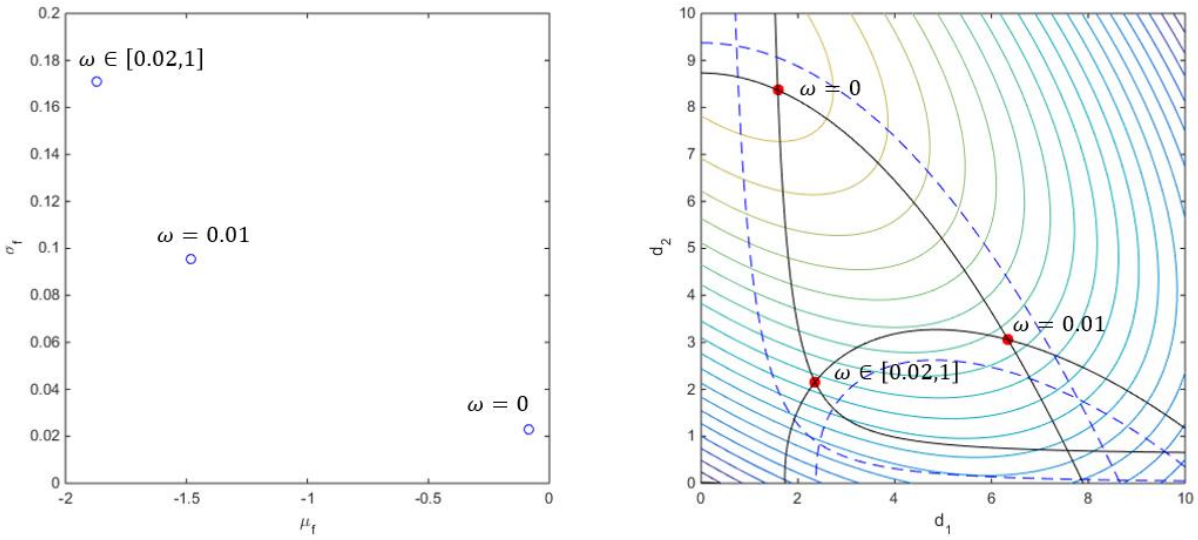


Fig. 1.34. Results for Formulation 5

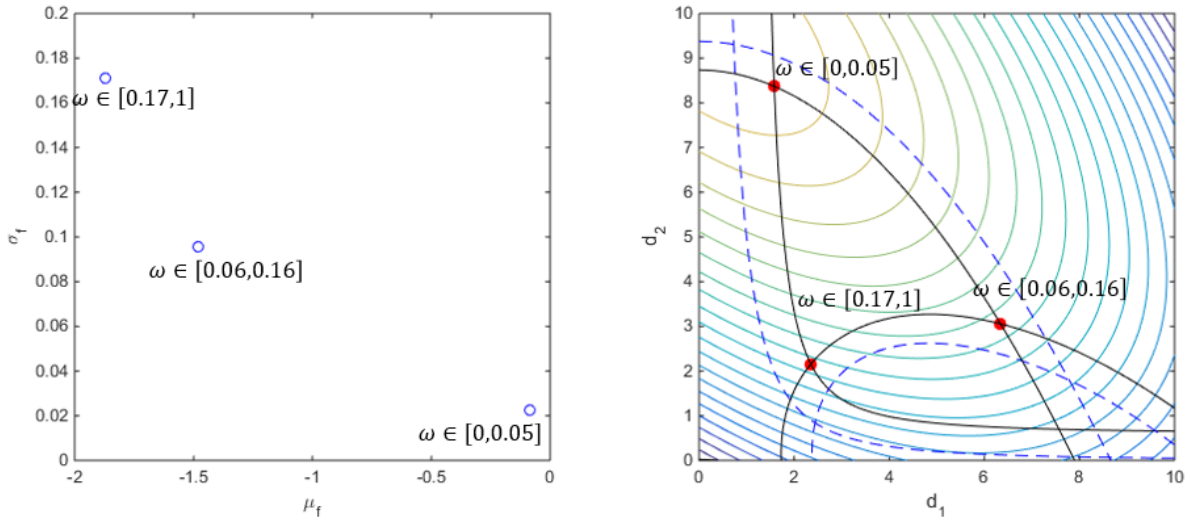


Fig. 1.35. Results for Formulation 6

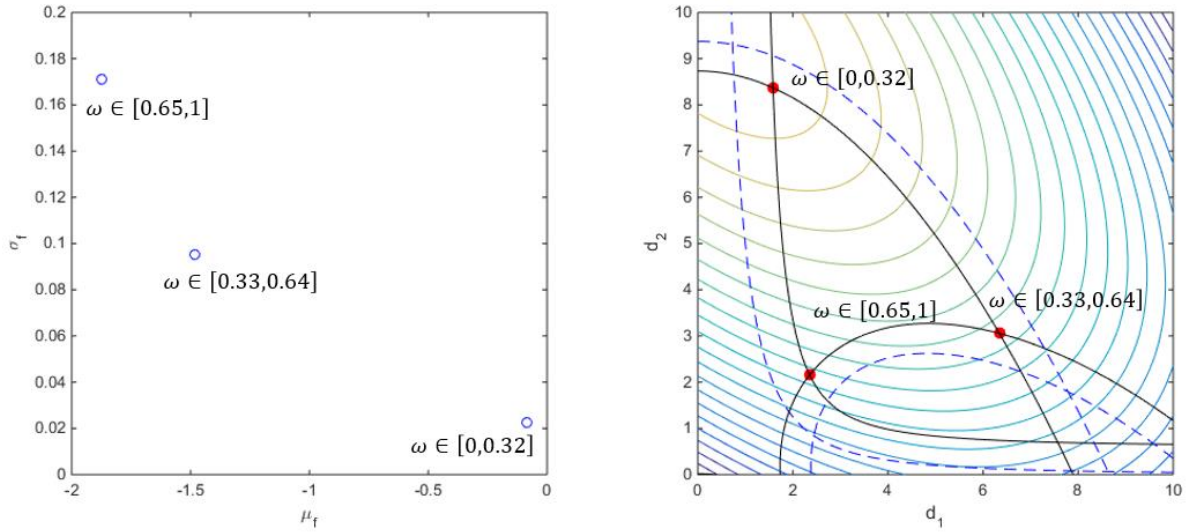


Fig. 1.36. Results for Formulation 7

The main divergence among the results of these three formulations is that the values of ω for obtaining each point are very different, as shown in following table.

For Formulation 5, as the square values of mean and standard deviation increase the difference of their orders of magnitudes, only when $\omega = 0$ or $\omega = 0.01$ the method could converge to the two former solutions, all the rest values of ω have no influence on the solution. Formulation 6 without the square information has more chance to obtain the two former solutions but more than 80% values of ω will result in the third one. Formulation 7 seems to solve the problem quite good by scaling the mean and standard deviations. Each solution is found by almost one third of ω values. This formulation is more balanced than the two other.

As Formulation 8 and 9 are multi-objective problems, SQP cannot solve them, so goal-attaining method and Non-dominated Sorting Genetic Algorithm (NSGA2) are both tested here.

The goal-attainment method needs a given coordinate and seeks to find a design belong to the Pareto front which is the closest point to the given coordinate following a given direction (Barba, 2009; Collette & Siarry, 2011). In this example, the goal is given as $[\mu_{fmin}; \sigma_{fmin}]$ for Formulation 8 and $[\mu_{fmin}/|\mu_f{}^*|; \sigma_{fmin}/\sigma_f{}^*]$ for Formulation 9. Also, the searching direction is changed from 0 to 1 with 100 equal sections.

NSGA2 is a global evolutionary algorithm that can treat more than one objective. The results of these two algorithms are shown in the following figures. Fig. 1.37 and 1.38 present results of Formulation 8 using goal-attainment and NSGA2 method respectively. Fig 1.39 and 1.40 show results of Formulation 9 with scaling values.

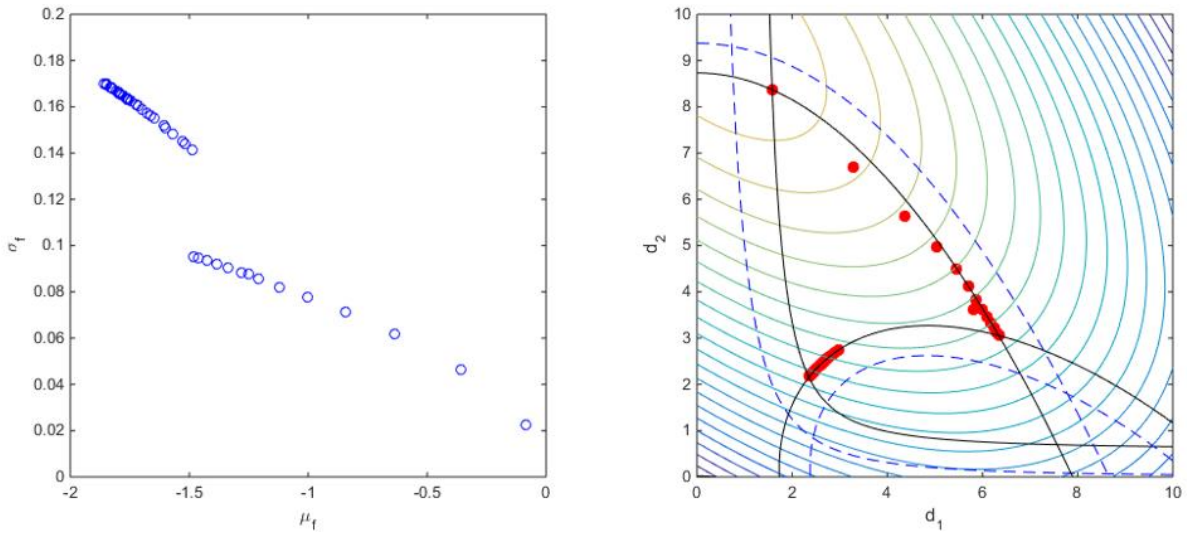


Fig. 1.37. Results for Formulation 8 using goal-attainment

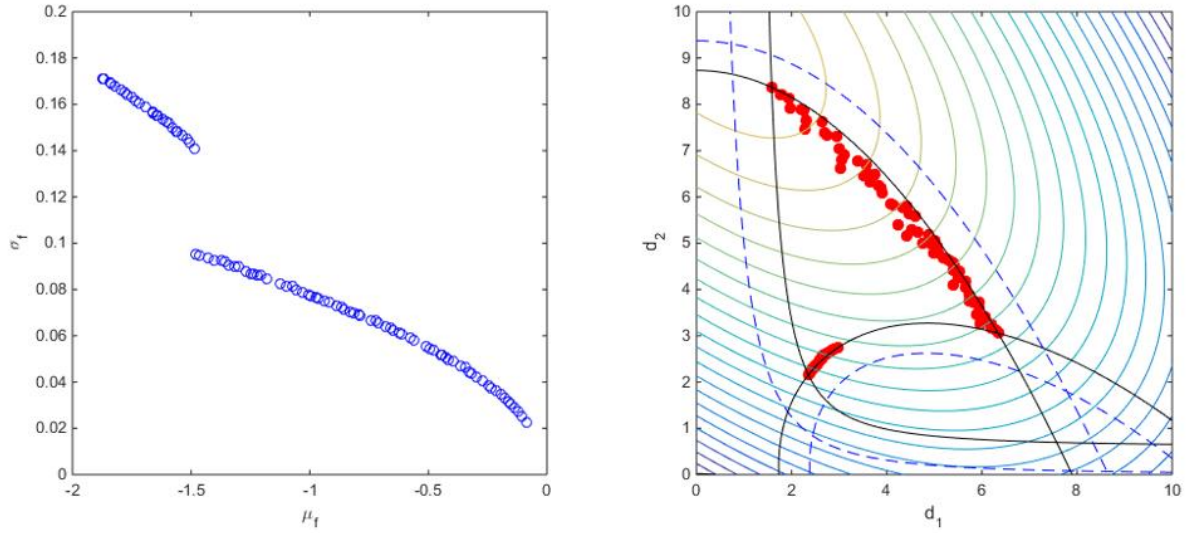


Fig. 1.38. Results for Formulation 8 using NSGA2

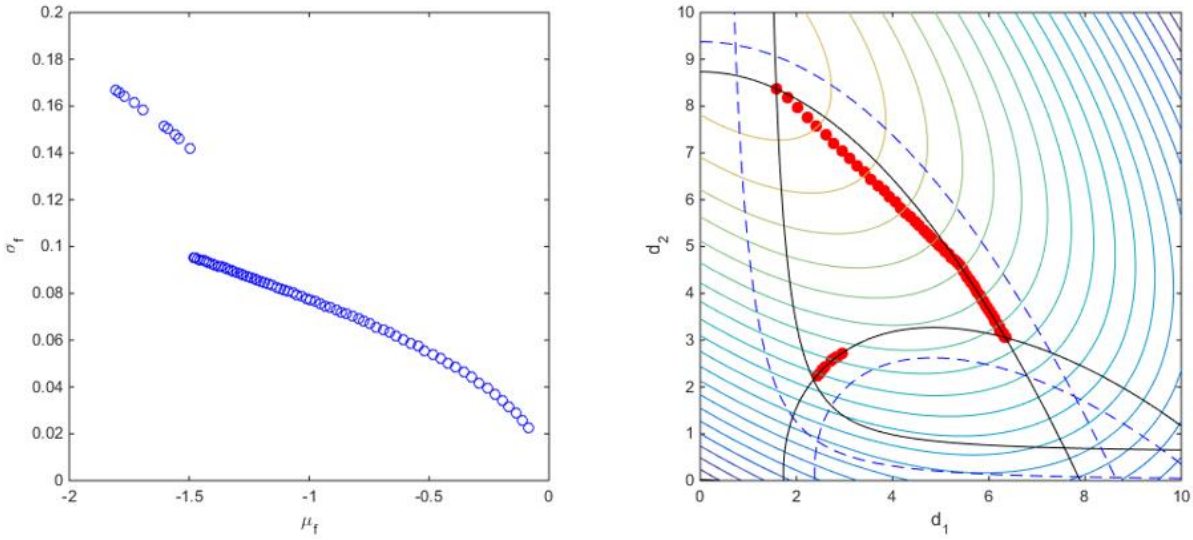


Fig. 1.39. Results for Formulation 9 using goal-attainment

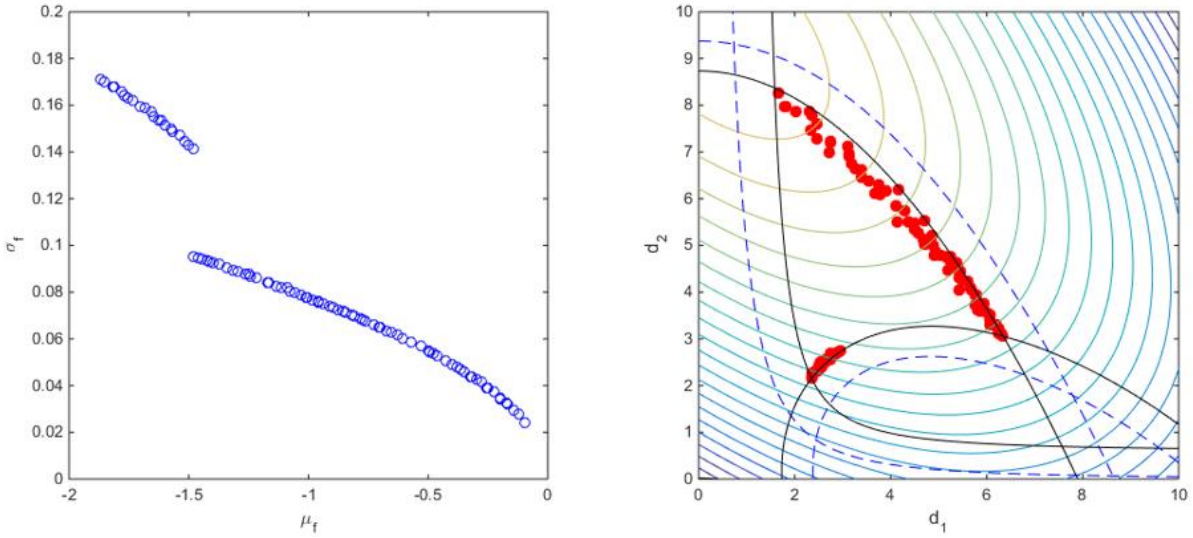


Fig. 1.40. Results for Formulation 9 using NSGA2

From the Fig 1.37 to 1.40, it can be seen that the algorithm goal-attainment depends on the formulation: without scaled values the result is not good enough as the distance between two adjacent solutions is large. NSGA2 does not have this problem but the solutions have disturbances and do not lie on a straight line.

A comparison for all 9 formulations is presented in the following table. The maximum and minimum solutions they can obtain are taken into account. Besides the maximal and minimal distances between two adjacent points found are also considered. It should be noticed that if the minimal distance is less than 10^{-4}, we treat it as 0 and maximal distance for the formulations which have more than 3 solutions is the distance except the gap of discontinuity. "Function evaluations" is the average number for one solution.

Table 1.6. Comparison of different formulations of RDO

Formulation	Minimal μ_f	Maximal μ_f	Minimal σ_f	Maximal σ_f	Minimal distance	Maximal distance	Function evaluations
1 with SQP	-1.8712	--	--	0.1712	--	--	1003
2 with SQP	-1.8712	--	--	0.1712	--	--	952
3 with SQP	-1.4829	--	--	0.0955	--	--	1598
4 with SQP	-1.8709	-0.0864	0.0228	0.1712	0	0.1551	1472
5 with SQP	-1.8712	-0.0862	0.0228	0.1712	4.0895	7.1320	1071
6 with SQP	-1.8712	-0.0862	0.0228	0.1712	4.0895	7.1320	1102
7 with SQP	-1.8712	-0.0862	0.0228	0.1712	4.0895	7.1320	1109
8 with goal-attain	-1.8619	-0.0864	0.0228	0.1706	0	2.3767	5498
8 with NSGA2	-1.8704	-0.0865	0.0228	0.1710	0	0.4865	12138
9 with goal-attain	-1.8692	-0.0864	0.0228	0.1711	0	0.3058	6883
9 with NSGA2	-1.8556	-0.0871	0.0229	0.1709	0.0001	0.4326	11862

Obviously, Formulation 1 and 2 can obtain the exact μ_{fmin} and σ_{fmax}, however as mono-objective methods, they could not change the solution with different way of trade-off between the mean and the standard deviation. Formulation 5, 6 and 7 can find the points which can achieve μ_{fmin} with σ_{fmax} and μ_{fmax} with σ_{fmin}, among them Formulation 7 is the best as the intervals of weight values lead to the three different points are almost equal. Formulation 4 with ϵ constraint and Formulations 8, 9 with multi-objective algorithms can obtain a Pareto front which presents all non-dominated solutions. However, the extrema found are not the real minimum and maximum for μ_f and σ_f exactly. Among all of them, Formulation 4 is the most effective as its solutions are closest to the real extremes and the distances between each two neighbouring solutions are the shortest. It is to mention that the average number of evaluations for Formulation

4 is almost the same as other formulations which could not obtain a complete Pareto front and at least 5 times less than Formulation 8 and 9 that found similar solutions.

2.3 Comparison of RBDO methods

As mentioned before, there are three categories of RBDO methods. For understanding the implications of the choice of different method and to quantify the impact of the choice of approaches and algorithms on the accuracy and the number of evaluations, six mentioned methods (RIA, PMA, AMA, SLA, SORA and SAP) are used and the results are given in the next table. The different optima are compared with each other in Fig 1.41 where the dotted curves present contours of objective and constraint boundaries. Limit-state $g_i = 0$ are depicted as solid lines. In order to maintain the consistency, the targeted index β_t is kept equal to 2 so that the targeted probability of failure P_t for RBDO is $\Phi(-\beta_t) = 2.28\%$. P_f is computed by Monte-Carlo simulation with a sampling of 10^6 realizations.

In order to better understand the difference between RBDO and Deterministic Design Optimization (DDO) without uncertainty, result of DDO is also presented here. The original formulation of DDO is expressed as:

$$\begin{aligned} &\min_{\boldsymbol{d}} f(\boldsymbol{d}) \\ &s.t.\ \boldsymbol{g}(\boldsymbol{d}) \leq \mathbf{0} \\ &\boldsymbol{d^L} \leq \boldsymbol{d} \leq \boldsymbol{d^U} \end{aligned} \tag{1.69}$$

Here we could see that only the mean of design variables is considered in the optimization process and no more reliability analysis or propagation of uncertainty are adjoined. As for double-loop method, the inner loop is also an optimization sub-problem. Thus beside the methods introduced in section 1.3 (HLRF for inner loop of RIA, HMV for inner loop of PMA) traditional optimization algorithms like SQP can also be suitable for the inner loop and these results are also presented here to compare each other for the sake of choosing the best method. Also, as mentioned in section 1.3.4, SAP method can use both principles of PMA and RIA, these two different SAP are also tested and presented here as SAP with PMA and SAP with RIA.

The column named convergence means that we run the algorithms 100 times with different initial points to see how many times it converges to the global optimum.

For this fast model example, from Table 1.7 and Fig 1.41, it is obviously that only the result of DDO lies on the limit-states and for that reason, the maximal P_f is 50% if the design variables follow the normal law. Most of the RBDO methods could find a result that satisfies the reliability targeted index. The probabilities of failure of RBDO methods are close to the targeted probability except for the single-loop method AMA because of the approximation used to increase the speed of convergence by sacrificing the accuracy.

Table 1.7. Results of the different methods of RBDO

Method	d_1	d_2	μ_f	σ_f	Max of P_f(%)	Convergence (100 runs)	Function evaluations
DDO	2.4398	0.8400	-2.6266	0.2072	50	20%	84
RIA with SQP	2.2513	1.9691	-1.9948	0.1792	2.28	29%	4539
RIA with HLRF	2.2513	1.9691	-1.9945	0.1792	2.28	33%	1729
PMA with SQP	2.2513	1.9691	-1.9945	0.1792	2.28	82%	3183
PMA with HMV	2.2513	1.9691	-1.9945	0.1792	2.28	84%	1318
AMA	2.3524	2.1568	-1.8712	0.1715	1.87	46%	132
SLA	2.2512	1.9677	-1.9953	0.1793	2.29	32%	165
SORA	2.2513	1.9691	-1.9945	0.1792	2.28	45%	531
SAP with RIA	2.2513	1.9691	-1.9945	0.1792	2.28	22%	278
SAP with PMA	2.2513	1.9691	-1.9945	0.1792	2.28	37%	181

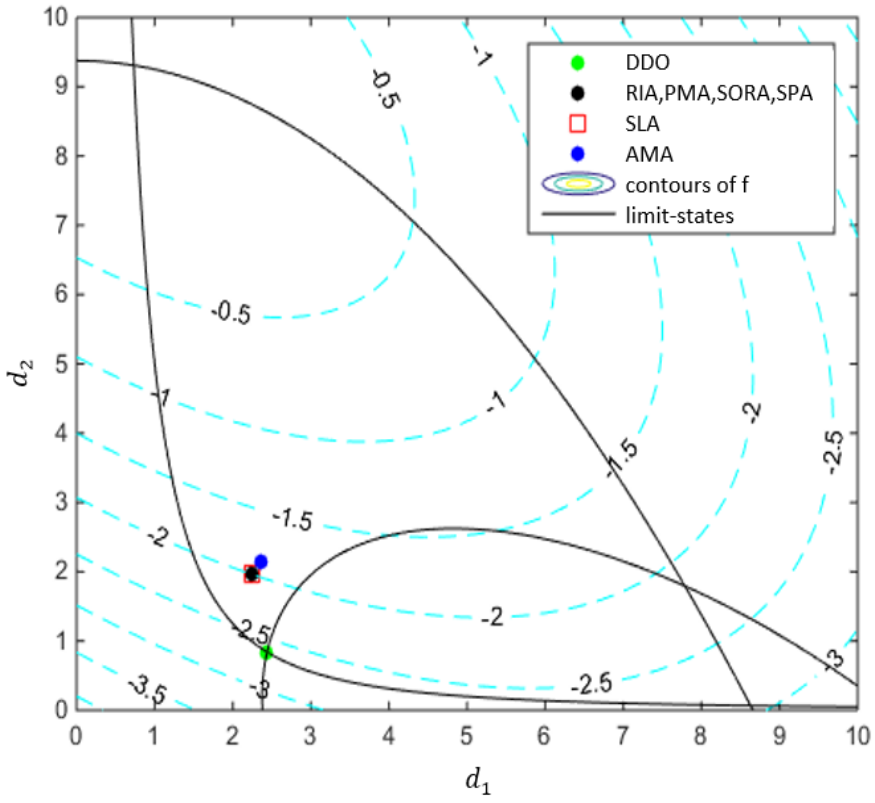


Fig. 1.41. The results of different methods plotted in the search space

For double-loop methods, the performance of using SQP for the inner loop is notable less than using HLRF or HMV, the difference is reflected in the need of more evaluations numbers. For PMA and RIA, SQP will need almost 3 times more evaluations than HMV and

HLRF, respectively. For these two methods, PMA is more effective not only due to the less number of evaluations but also because of it has larger convergence rate than RIA.

For single-loop methods, AMA is the fastest but the most inaccurate. SLA is slower than AMA but still faster than any other RBDO methods, besides it is more accurate than AMA although it still has a little difference with others. It presents a good compromise between accuracy and speed.

Sequential decoupled methods are faster than double-loop methods and slower than single-loop ones but they do not loss accuracy. Among them, SAP is the most effective.

Among all RBDO methods, we can conclude that for this fast model example, SAP is the most efficient method as it can obtain the solution without losing any accuracy and a number of evaluations close to single-loop methods.

2.4 Comparison of RBRDO methods

The three types of RBRDO (1: Normal-the-best, 2: Smaller-the-better, and 3: Larger-the-better) are used here with two other formulations presented by (Ziyan Ren et al., 2013) and (Paiva, 2010). The synthesis is shown in the following table.

Table 1.8. Different formulations used for RBRDO

Formulation	Objective function	Constraint
1	$f_r(\mu_f, \sigma_f) = \omega_1 \left(\frac{\mu_f - f_t}{\mu_{f_0} - f_t}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$P_f(g(\boldsymbol{x}) > 0) \leq P_t$
2	$f_r(\mu_f, \sigma_f) = \omega_1 \cdot \text{sign}(\mu_f) \left(\frac{\mu_f}{\mu_{f_0}}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$P_f(g(\boldsymbol{x}) > 0) \leq P_t$
3	$f_r(\mu_f, \sigma_f) = \omega_1 \cdot \text{sign}(\mu_f) \left(\frac{\mu_{f_0}}{\mu_f}\right)^2 + \omega_2 \left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$	$P_f(g(\boldsymbol{x}) > 0) \leq P_t$
4	$f_r(\boldsymbol{d}) = f_w(\boldsymbol{d})$	$P_f(g(\boldsymbol{x}) > 0) \leq P_t$
5	$f_r(\boldsymbol{d}) = \mu_f(\boldsymbol{d}) + \sigma_f(\boldsymbol{d})$	$P_f(g(\boldsymbol{x}) > 0) \leq P_t$

P_t is 2.28% in order to keep the consistency with β_t. ω_1, ω_2 are both equal to 0.5, μ_{f_0} and σ_{f_0} are calculated by propagating the uncertainties of the chosen starting point in the beginning, in this example, we fixed it as the value computed by the design point [3.5; 5]. f_t is the targeted value which we expect, as at the beginning the minimum of RBRDO is unknown, so we use the minimum of DDO to instead. However it should be noticed that this value $f_t = -2.6266$ is just an ideal result and could never be attain as the constraints change. $f_w(\boldsymbol{d})$ in Formulation 4 is the worst-case of design point $\boldsymbol{d}$ in the uncertainty set defined in Equation (1.2) and can be calculated in the same way as presented in WWCO.

Results are given in Table 1.9 and Fig 1.42. It can be noticed that the average numbers of evaluations are similar and the results can almost satisfy the targeted probability of failure. Formulation 4 and Formulation 5 find exactly the same solution as RBDO because

the processing of constraints is made in the same way as RBDO. Formulation 4 does not consider the standard deviation of objective and Formulation 5 takes much more attention on the mean than the deviation as their magnitude are different.

Moreover, with the same weight factor ω, STB (Formulation 2) could find the smallest μ_f that corresponds to its purpose. The value of μ_f found by NTB (Formulation 1) is larger than STB and LTB (Formulation 3) finds the largest μ_f as expected.

Table 1.9. Results given by different formulations of RBRDO

Formulation	1	2	3	4	5
x_{opt}	[5.6590; 4.3016]	[6.4706; 2.9672]	[1.3675; 8.5018]	[2.2513; 1.9691]	[2.2513; 1.9691]
$\mu_f(x_{opt})$	-1.0750	-1.5301	-0.0690	-1.9945	-1.9945
$\sigma_f(x_{opt})$	0.0803	0.0968	0.0206	0.1789	0.1789
$P_{f1}(\%)$	0	0	2.2778	2.2782	2.2782
$P_{f2}(\%)$	1.4053	2.1685	0	2.2841	2.2841
$P_{f3}(\%)$	2.2761	2.2952	2.3045	0	0
Evaluations	1714	1612	1746	1510	1802

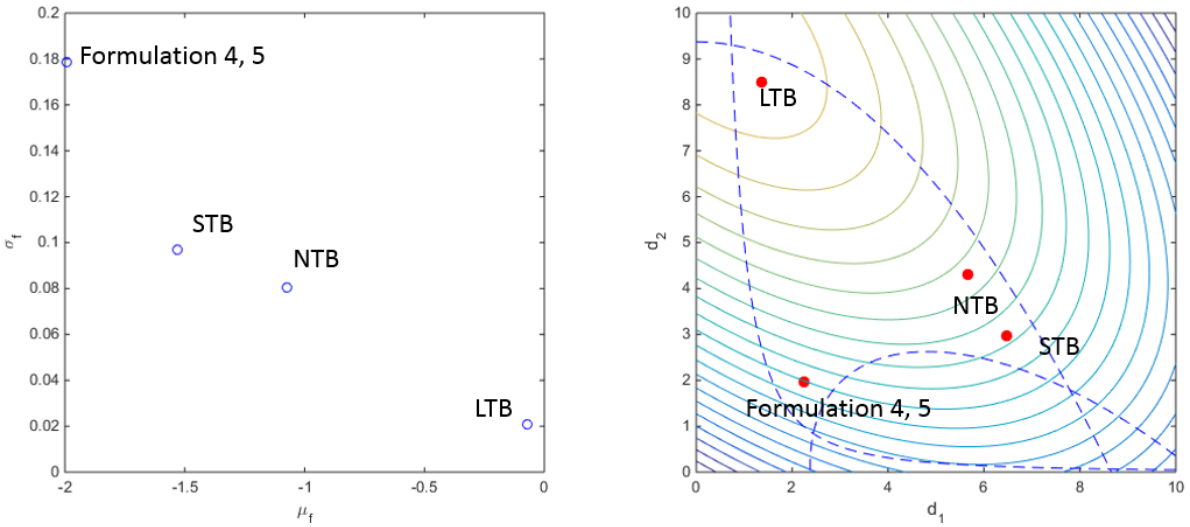


Fig. 1.42. The results of different RBRDO methods

Besides, if we change the value of weight parameters ω_1 and ω_2, the former three formulations give normally a Pareto front. However, due to the different formulations, the results are also different. For NTB we move ω_1 from 0 to 1 with step 0.01 and let $\omega_2 = 1 - \omega_1$, it could find various solutions as in Fig 1.43 but for STB it can only reach the three bounds and for LTB only one solution can be found.

In fact, the results of the Formulation 2 to 5 in Table 1.9 are the three bounds of the Pareto front. It means all these results can be obtained by the formulation Normal-the-best by changing the weight parameter. So it can be concluded that for this example, NTB is the most universal formulation as it can contain all results of other formulations.

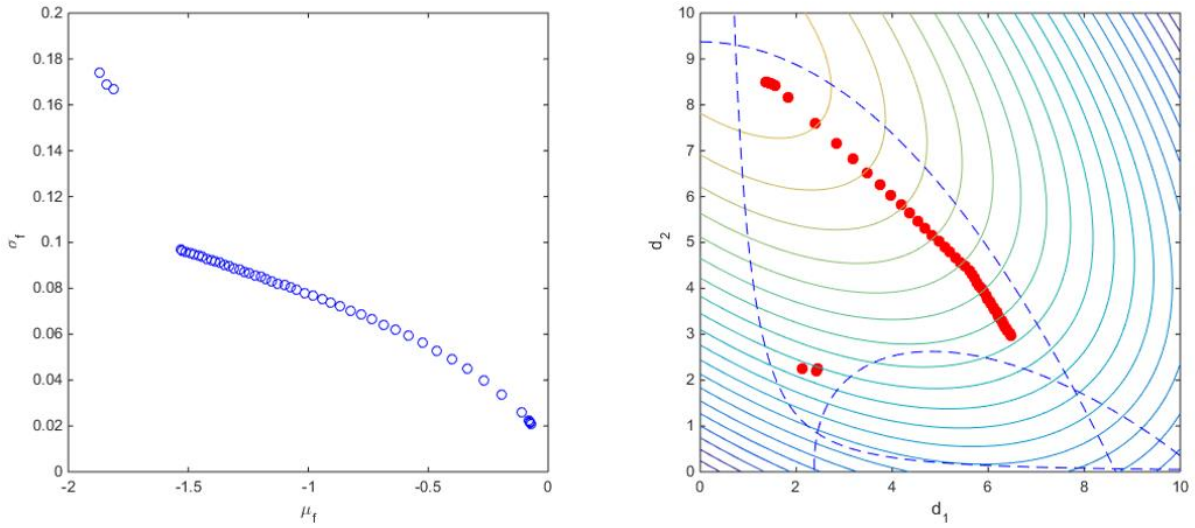


Fig. 1.43. Results for NTB (Formulation 1) with different ω

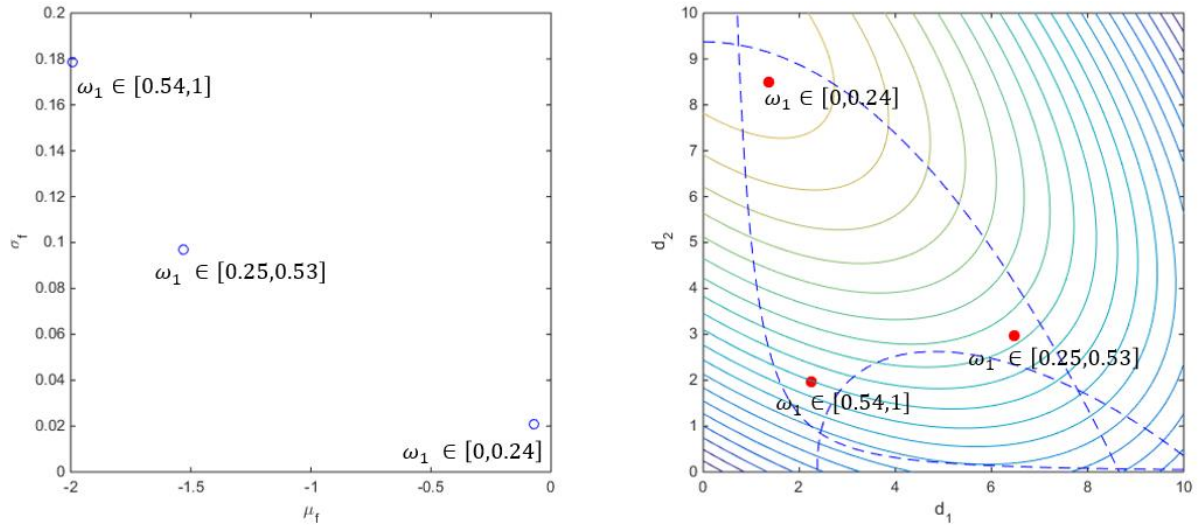


Fig. 1.44. Results for STB (Formulation 2) with different ω

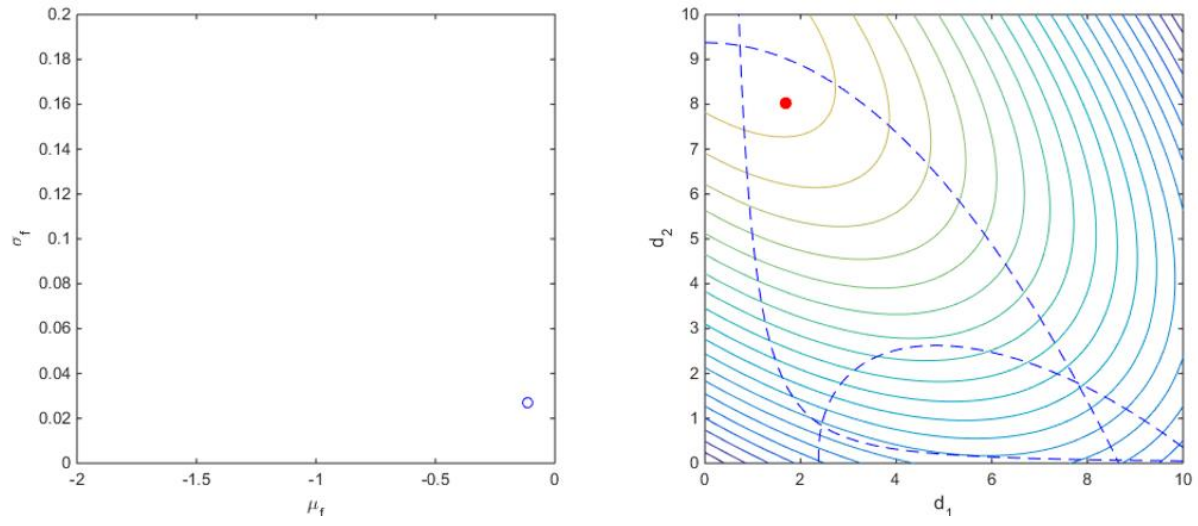


Fig. 1.45. Results for LTB (Formulation 3) with different ω

An interesting thing is that as Formulation 7 which uses also scaled values in RDO, STB could only find the bounds of the Pareto front as the relation function of μ_f and σ_f is concave. However, for NTB it can attain a Pareto front better than STB. The reason is that the NTB focus on $\left(\frac{\mu_f - f_t}{\mu_{f_0} - f_t}\right)^2$ and $\left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$, but the relationship of these two terms are no longer concave, it turns to convex instead. The following figure presents the relations between μ_f and σ_f, $\frac{\mu_f}{|\mu_{f_0}|}$ and $\frac{\sigma_f}{\sigma_{f_0}}$ from RDO, $\text{sign}(\mu_f)\left(\frac{\mu_f}{\mu_{f_0}}\right)^2$ and $\left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$ from STB, $\left(\frac{\mu_f - f_t}{\mu_{f_0} - f_t}\right)^2$ and $\left(\frac{\sigma_f}{\sigma_{f_0}}\right)^2$ from NTB respectively and it can be noticed that the last function type is different from the formers because in this analytical example, the value of initial objective function is always negative so that the square operation changes the relation.

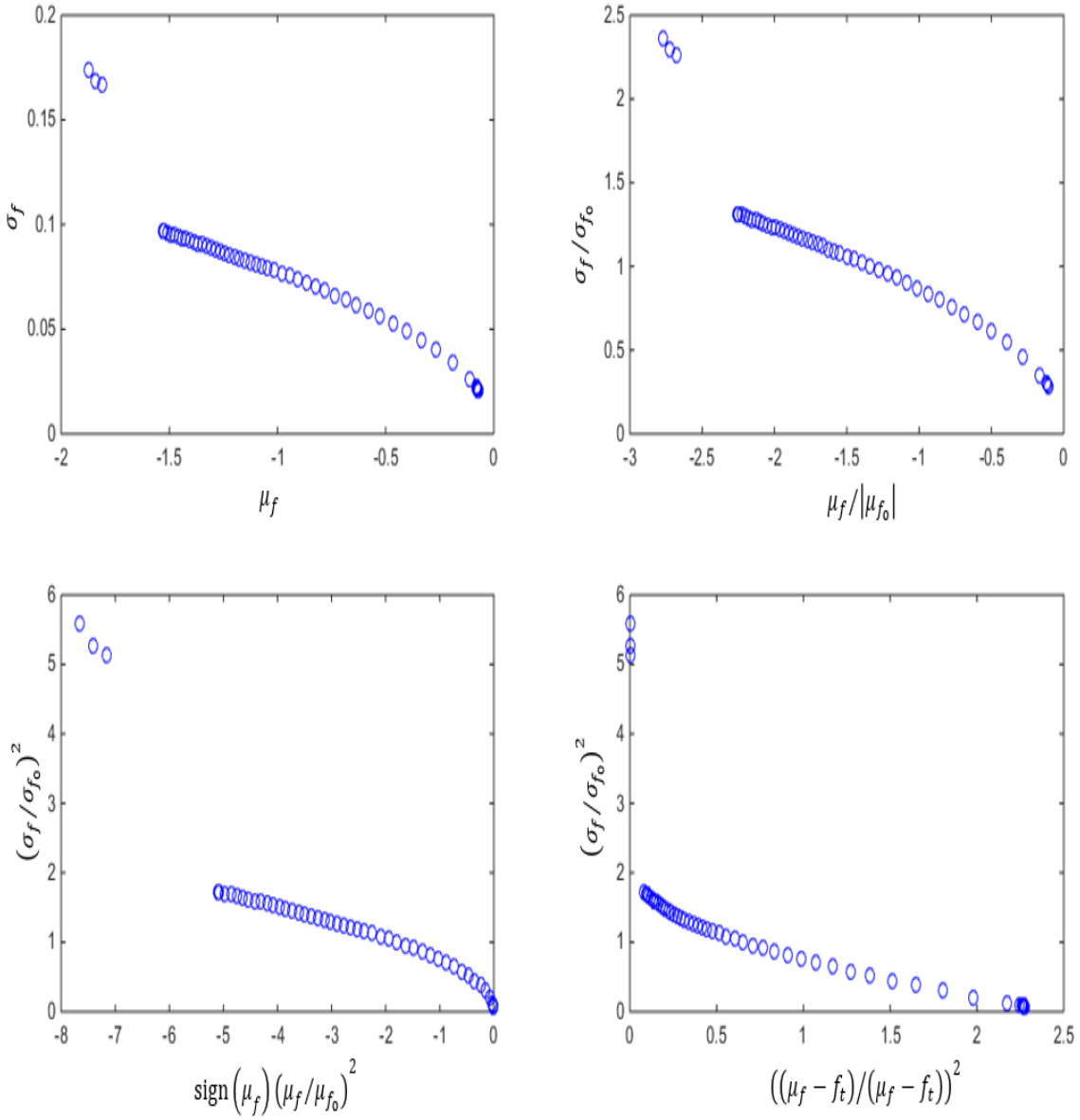


Fig. 1.46. Different relation functions

3 Conclusion

All of the four different categories taking into account uncertainty (WCO, RDO, RBDO and RBRDO) aim at finding a more robust or reliable solution but their principals are not the same. The following table synthesizes the differences among the four categories on the parts of input information, objective, and strategies.

Table 1.10. A comparison among optimization methods considering uncertainty

Method	WCO	RDO	RBDO	RBRDO
Information on input	Mean, standard deviation or gradient	Mean and standard deviation	PDF	PDF
Objective	Minimization and variability reduction	Minimization of mean and/or standard deviation of objective	Minimization under probabilistic constraints	Minimization of mean and/or standard deviation of objective under probabilistic constraints
Analysis type	Variation analysis	uncertainty propagation	Reliability analysis	uncertainty propagation and reliability analysis
Strategy	moving the mean	consider two objectives	Moving the mean	consider two objectives and moving the mean

This chapter introduces separately these four different categories of optimizations that can deal with input uncertainties and presents the numerical investigation of these methods on a mathematical example.

Each category has several methods or formulations to tackle the uncertainty and to balance their objectives, but their performances are unequal.

For WCO methods, with the fast mathematical model, WWCO seems more efficient than traditional WCO approach or gradient based approaches.

For RDO, as they consider both the mean and standard deviation of initial objective function, multi-objective formulations or those with weight parameters and scaled values are more suitable as they can give different reference results with different emphasis.

RBDO can be derived into three type of methods, single loop methods are the fastest but they lose the accuracy, double-loop methods need the largest number of evaluations, sequential decoupled methods are the most efficient. Among those last methods SAP is the most suitable method on the analytical example.

RBRDO yields the same conclusion as RDO: methods with scaled values and weight parameters are more suitable and, among them, Normal-the-best formulation is the most universal.

However, these methods can only be applied with fast models. The biggest difference between fast model and cumbersome model like finite element models used in electromagnetic, is that the latter one is time-consuming to evaluate so that optimization will be very time costly. Thus, the formulations and optimization methods presented in this chapter become unbearable because of the huge time consumption. For this reason, more efficient methods which use surrogate models and could combine modelling and optimization together in the save process are promising. The next chapter will introduce the robust or probabilistic optimization methods adapted to heavy models.

Chapter 2: Methods adapted to heavy models

1 State of the art

Nowadays numerical simulations like Finite Element Analysis (FEA) are playing a more and more important role in electromagnetic device designs as we can use them to simulate complicated geometry systems with high accuracy.

However, the electromagnetic devices are sometimes too complex and even the simulations of these devices' behaviors need heavy analysis. Hence, single evaluation of FEA could take several minutes up to even days. Therefore, the expense of running analysis is non-negligible. For the sake of avoiding a significant overall time in the optimization process, meta-models have gained more and more attention. The idea of meta-model based optimization is using fast-evaluation meta-models to replace the high-fidelity heavy models in order to reduce the computational burden of optimizations.

Besides, other times, we do not know what is inside the high-fidelity model. That means the system or device is a black box for which we only know the input and output relationships without knowing the internal structure. To optimize with black-box models that are computationally expensive, the meta-model should also be built in advance.

In addition to these mentioned necessities, there are other advantages of using meta-models. Numerical models are sources of numerical noise for example the numerical noise of finite element analysis (FEA) introduced by the mesh adaptation (Mester, 2007; Neittaanmäki, Rudnicki, & Savini, 1996). Thus, if a gradient-based algorithm is used, the convergence may be affected. However, meta-models are usually noise-free models (sufficiently smooth function with respect to the inputs) so that this problem do not occur.

However, there are also some drawbacks of using meta-model based optimization. First, it should be noticed that the global prediction accuracy of meta-models is always inferior to the high-fidelity finite element models or the real system even when a high number of sampling points are chosen to construct the meta-model. The precision and the reduction of computational burden cannot be obtained simultaneously. Thus, our proposition is to use a limited sample size to improve the accuracy of meta-model as much as possible. One of the effective methods is to continually reconstruct the meta-model during the optimization design process according to the requirements. This approach is called adaptive meta-model and it can add new samples at the most needed place to increase the accuracy gradually.

This chapter starts by introducing some common methods for creating meta-models and some mostly used adaptive meta-model based optimizations with infill searching criteria (ISC). Then, different types of meta-model based optimization strategies for heavy models are presented in details and the most efficient strategies for each type of design optimization are highlighted.

1.1 Surrogate models

Optimization process can be greatly accelerated by using suitable meta-models, also called surrogate models, to replace the high-fidelity heavy model, even if at the expense of a loss in accuracy (Rijpkema, Etman, & Schoofs, 2001). To sum-up in a sentence, the surrogate model is using some approaches to create a continuous function from a discrete sample of assessment points (Bompard, 2011). From the early 50s of the last century, many different methods to create surrogate models were proposed. Most of them are based on interpolation. Initially, we find of course the methods of linear interpolation, then quadratic. Quadratic interpolation involves a second order polynomial to fit the function and it is fast in both creation and evaluation (Giunta & others, 1997). These methods were then extended to polynomial interpolation, it is the basis of the Response Surface Methodology (RSM) proposed by (Gep & Kb, 1951). And also, it has been used for regression methods.

In the late 1950s, (Lettvin, Maturana, McCulloch, & Pitts, 1959) first theorized the Artificial Neural Networks (ANN), which is considered as the basis of neural networks. They demonstrated in particular that neural networks can approximate complex logical, algebraic or symbolic functions. In the following years, ANN were studied in great details and nowadays there are over 20 different types of ANN used in a vast range of applications ranging from non-linear control to data mining (Liu & Batill, 2000; Paiva, 2010).

Since 1980s, three classes of meta-model methods have been proposed and because of their simplicity of construction they have been widely used. The first class is Multivariate Adaptive Regression Splines (MARS) proposed by (Friedman, 1991). This technique is a two-pass stepwise non-parametric regression and can be considered as an extension of linear models that automatically models nonlinearities and interactions between variables. The second class is Radial Basis Function (RBF) which uses a real-valued function whose value depends only on the distance from the origin or alternatively on the distance from some other point. (Broomhead & Lowe, 1988) proposed to use this type of function in neural networks and since then with the contributions of (Moody & Darken, 1989), the RBF network has become widespread and alternative to traditional neural network due to the latter's expensive cost of training models (Orr & others, 1996). Last class is Kriging model, which was first developed in geostatistical (Krige, 1951) and then extended to interpolation of mathematical functions (Sacks, Welch, Mitchell, & Wynn, 1989).

In the 1990s, a new method called Support Vector Machine was proposed (Cherkassky, 1997). Numerous researches have been devoted to the development of this type of method in recent years. However, the applications to physical problems seem quite few.

Comparative studies have been made between many methods of surrogate modelling. (Sacks et al., 1989) compared Kriging to polynomial interpolation methods in aeronautical applications, (Jin, Chen, & Simpson, 2001) and (Peter, Marcelet, Burguburu, & Pediroda, 2007) compared polynomial regression, Kriging, MARS, and RBF in mathematical models and industrial applications. (Varadarajan, Chen, & Pelka, 2000) compared neural networks with polynomial regression in thermodynamic applications. Among these comparisons, there is not one method that surpasses all others in all applications, nevertheless, it seems that Kriging is generally the most effective one (Jin, Du, & Chen, 2003). Therefore, this method will be presented in more details in next section.

1.2 Kriging

Kriging is a statistical interpolation method. In the 1960s, his work was then mathematically formalized by Matheron at Ecole des Mines de Paris (Matheron, 1963). It was in the late 1980s that (Sacks et al., 1989) proposed to use this method for the approximation of functions from computer experiments and since then the use of Kriging has developed quickly and numerous variants have been proposed (Currin, Mitchell, Morris, & Ylvisaker, 1991). (Huang, 2005; Lee & Park, 2006; Simpson, Mauery, Korte, & Mistree, 2001) gave some recent examples of using Kriging to develop surrogate models in design optimization.

Kriging model makes it possible to use the determination of small size sample to approach complex functions, it estimates the value of the function to be predicted at some given point by a linear combination of learning samples. The weight associated with each sample depends only on the distance at the point considered, and it is assumed that points which are close to each other in the space tend to have similar characteristics so that the nearest sample will have the largest weight value (Baudoui, 2012).

As Kriging is based on Gaussian process, the latter will be introduced first.

1.2.1 Gaussian process

Gaussian process (GP) is a generalization of the multivariate normal distribution and defines a probability distribution over functions in the continuous domain. In a Gaussian process, every point in some continuous input space is associated with a normally distributed random variable and every finite collection of those random variables has a multivariate normal distribution. The distribution of a Gaussian process is the joint distribution of all those random variables (Rasmussen & Williams, 2006).

A GP is determined by a mean $\mu_Z(\boldsymbol{x})$ and a kernel that is also known as covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$:

$$Z \sim GP(\boldsymbol{\mu_Z}, K): \mu_Z(\boldsymbol{x}) = E\big(Z(\boldsymbol{x})\big), K(\boldsymbol{x}, \boldsymbol{x}') = Cov\big(Z(\boldsymbol{x}), Z(\boldsymbol{x}')\big) \tag{2.1}$$

where $\boldsymbol{x}$ and $\boldsymbol{x}'$ are two input points.

From the definition, we can see that any set of N input vectors $\boldsymbol{X} = \big(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(N)}\big)$ associates with a N dimensional vector $\boldsymbol{Z} = \Big(Z\big(\boldsymbol{x}^{(1)}\big), Z\big(\boldsymbol{x}^{(2)}\big), \dots, Z\big(\boldsymbol{x}^{(N)}\big)\Big)$ that has a multivariate Gaussian distribution:

$$\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{\mu_Z}, \boldsymbol{C}): \boldsymbol{\mu_Z} = \begin{bmatrix} \mu_Z\big(\boldsymbol{x}^{(1)}\big) \\ \vdots \\ \mu_Z\big(\boldsymbol{x}^{(N)}\big) \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} K\big(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(1)}\big) & \cdots & K\big(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(N)}\big) \\ \vdots & \ddots & \vdots \\ K\big(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(1)}\big) & \cdots & K\big(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(N)}\big) \end{bmatrix} \tag{2.2}$$

If we want to calculate the probability distribution function at some other points for example $\{\boldsymbol{X}^*, \boldsymbol{Z}^*\}$, while the information of $(\boldsymbol{X}, \boldsymbol{Z})$ is given, the normal distribution of vector $\boldsymbol{Z}^*$ can be written as $\boldsymbol{Z}^* \sim \mathcal{N}(\boldsymbol{\mu_Z}^*, \boldsymbol{C}^{**})$ where $\boldsymbol{C}^{**} = K(\boldsymbol{X}^*, \boldsymbol{X}^*)$. So that the joint probability function of $\boldsymbol{Y}$ and $\boldsymbol{Z}^*$ is:

$$\begin{bmatrix} \boldsymbol{Z} \\ \boldsymbol{Z}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu_Z} \\ \boldsymbol{\mu_Z}^* \end{bmatrix}, \begin{bmatrix} \boldsymbol{C} & \boldsymbol{C}^* \\ \boldsymbol{C}^{*\boldsymbol{T}} & \boldsymbol{C}^{**} \end{bmatrix}\right) \tag{2.3}$$

where $\boldsymbol{C}^* = K(\boldsymbol{X}, \boldsymbol{X}^*)$. Therefore the conditional distribution of $\boldsymbol{Z}^*$ given $\boldsymbol{Z}$ can be expressed as (Brochu, Cora, & De Freitas, 2010):

$$p(\boldsymbol{Z}^*|\boldsymbol{Z}) \sim \mathcal{N}(\boldsymbol{\mu_Z}^* + \boldsymbol{C}^{*T}\boldsymbol{C}^{-1}(\boldsymbol{Z} - \boldsymbol{\mu_Z}), \boldsymbol{C}^{**} - \boldsymbol{C}^{*\boldsymbol{T}}\boldsymbol{C}^{-1}\boldsymbol{C}^*) \tag{2.4}$$

From the definition of GP, we know that the realization of the GP is determined by the mean and the kernel (covariance). Therefore, how to select a fitted kernel which can represent the structure present in the data is one of the main issue. Since covariance functions are closed under addition and multiplication, that is to say, if C_1 and C_2 are two covariance functions of random process with a common parameter set, then $\alpha_1 C_1 + \alpha_2 C_2$ and $C_1 \cdot C_2$ are also covariance when $\alpha_1, \alpha_2 > 0$ (Todorovic, 2012). With this property, it is possible to create sophisticated structures through combining them (Duvenaud, Nickisch, & Rasmussen, 2011). Several functions have been proposed for this modeling (Kleijnen, 2009). Examples can be found in next table where ${\sigma_Y}^2$ is the variance of Y and θ, θ' are the parameters.

Table 2.1. Examples of covariance models for GP correlation model

Covariance function	Expression
Exponential	${\sigma_Y}^2 exp(-(\|\boldsymbol{d} - \boldsymbol{d}'\|)/\theta)$
Power Exponential	${\sigma_Y}^2 exp\big(-((\|\boldsymbol{d} - \boldsymbol{d}'\|)/\theta)^{\theta'}\big)$ where $0 < \theta' \leq 2$
Squared Exponential	${\sigma_Y}^2 exp(-(\|\boldsymbol{d} - \boldsymbol{d}'\|)/(2\theta^2))$

Note that the values of all the functions introduced in the table above depend only on the Euclidean distance between their input vectors, this is called stationary. A stationary covariance function is said isotropic or homogeneous if it depends only on the Euclidean norm, otherwise it is called anisotropic.

Kriging is based on the assumption that a function $y: \mathbb{R}^p \to \mathbb{R}$ is a realisation of a conditional Gaussian process Y and can be decomposed into two parts: a deterministic part $\mu_Y(\boldsymbol{x})$ and a stochastic part $\epsilon(\boldsymbol{x})$:

$$Y(\boldsymbol{x}) = \mu_Y(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{2.5}$$

where $\boldsymbol{x} \in \mathbb{R}^p$, μ_Y represents the regression model also called kriging trend of the process, Z is a stationary Gaussian process $Z \sim GP(0, K)$, the expectation is zero $E[Z(\boldsymbol{x})] = 0$, and the variance ${\sigma_Y}^2 \in \mathbb{R}$ exist and the covariance is:

$$Cov\big(Z(\boldsymbol{x}), Z(\boldsymbol{x}')\big) = K(\boldsymbol{x}, \boldsymbol{x}') = {\sigma_Y}^2 R(\boldsymbol{x}, \boldsymbol{x}') \tag{2.6}$$

where R is a correlation function which depends only on the distance of the two points, and the covariance function can be chosen from the Table 2.1. And with the properties of covariance $K(\boldsymbol{x}, \boldsymbol{x}) = {\sigma_Y}^2$ and $K(\boldsymbol{x}, \boldsymbol{x}') = K(\boldsymbol{x}', \boldsymbol{x})$, we can also know that

$$\begin{cases} R(\boldsymbol{x}, \boldsymbol{x}) = 1 \\ R(\boldsymbol{x}, \boldsymbol{x}') = R(\boldsymbol{x}', \boldsymbol{x}) \end{cases} \tag{2.7}$$

In general, the value of a variable at a point is correlated to the values of the neighboring points, so that the establishment of a kriging model is to approximate the stochastic process Y at an unknown point $\boldsymbol{x}^*$ by a linear combination of the observed values. We assume that the expectation $\hat{y}$ of the realizations of this process is an approximation of the function Y. Suppose that $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(N)}\}$ is a set of N design sample points and $\boldsymbol{Y} = \{y(\boldsymbol{x}^{(1)}), y(\boldsymbol{x}^{(2)}), \dots, y(\boldsymbol{x}^{(N)})\}$ are the function values. The linear combination can be written as the following form:

$$\hat{y}(\boldsymbol{x}^*) = \sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) y(\boldsymbol{x}^{(i)}) \tag{2.8}$$

where λ_i are the respective weights which need to be estimated.

To estimate the average trend, the least square method can be used (Hansen, 2007). According to the Gauss-Markov theorem (Kruskal, 1968), in a linear regression model, the ordinary least squares is the Best Linear Unbiased Predictor (BLUP) only if it is unbiased and also it has the lowest mean square error of the estimate among all others. The properties can be written as:

1. $E\big(Y(\boldsymbol{x}) - \hat{y}(\boldsymbol{x})\big) = 0$;
2. $Var\big(Y(\boldsymbol{x}) - \hat{y}(\boldsymbol{x})\big) = E\left[\big(Y(\boldsymbol{x}) - \hat{y}(\boldsymbol{x})\big)^2\right] = \hat{s}^2(\boldsymbol{x})$, if this variance does not depend on the chosen point, that means error terms have the same variance called homoscedasticity

So the purpose is to find the estimator that can minimize the variance:

$$\hat{s}^2(\boldsymbol{x}^*) = Var\big(Y(\boldsymbol{x}^*) - \hat{y}(\boldsymbol{x}^*)\big) = Var\big(\hat{y}(\boldsymbol{x}^*)\big) + Var\big(Y(\boldsymbol{x}^*)\big) - 2Cov\big(\hat{y}(\boldsymbol{x}^*), Y(\boldsymbol{x}^*)\big)$$

$$= Var\left(\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) y_i\right) + Var\big(Y(\boldsymbol{x}^*)\big) - 2Cov\left(\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) y_i, Y(\boldsymbol{x}^*)\right)$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N} Cov\big(\lambda_i(\boldsymbol{x}^*) y_i, \lambda_j(\boldsymbol{x}^*) y_j\big) + Var\big(Y(\boldsymbol{x}^*)\big) - 2\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) Cov\big(y_i, Y(\boldsymbol{x}^*)\big) \qquad (2.9)$$

where $y_i = y\big(\boldsymbol{x}^{(i)}\big)$ and it is also unbiased:

$$E[\hat{y}(\boldsymbol{x}^*) - Y(\boldsymbol{x}^*)] = \sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) \mu_Y\big(\boldsymbol{x}^{(i)}\big) - \mu_Y(\boldsymbol{x}^*) = 0 \qquad (2.10)$$

There are three most well-known Kriging methods, their difference lies in the form of average trend $\mu_Y(\boldsymbol{x})$.

- Simple Kriging (SK): the trend is a known constant,
- Ordinary Kriging (OK): the trend is an unknown constant, $\mu_Y(\boldsymbol{x}) = \mu_Y$.
- Universal Kriging (UK): the trend is an unknown function, it can be written as a linear or quadratic combination of q known functions t_i, weighted with the coefficients $\beta_i \in \mathbb{R}$: $\mu_Y(\boldsymbol{x}) = \sum_{i=1}^{q} \beta_i\, t_i(\boldsymbol{x})$.

Each type of Kriging will be present in following sections.

1.2.2 Simple Kriging (SK)

For Simple Kriging, the trend is a known constant, $\mu_Y(\boldsymbol{x}) = 0$ for example. The purpose of Kriging is to use the existing samples to observe a realization of a stationary Gaussian process $Y(\boldsymbol{d}^*)$, so for Simple Kriging, with the help of deduction in (Lophaven, 2002), the prediction mean and variance at this point $\boldsymbol{d}^*$ are:

$$\begin{aligned} \hat{y}(\boldsymbol{x}^*) &= \mu_Y + \boldsymbol{c}(\boldsymbol{x}^*)^T \boldsymbol{C}^{-1}(\boldsymbol{y} - \boldsymbol{1}\mu_Y) = \mu_Y + \boldsymbol{r}(\boldsymbol{x}^*)^T \boldsymbol{R}^{-1}(y - \boldsymbol{1}\mu_Y) \\ \hat{s}^2(\boldsymbol{x}^*) &= {\sigma_Y}^2 - \boldsymbol{c}(\boldsymbol{x}^*)^T \boldsymbol{C}^{-1} \boldsymbol{c}(\boldsymbol{x}^*) = {\sigma_Y}^2\big(1 - \boldsymbol{r}(\boldsymbol{x}^*)^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}^*)\big) \end{aligned} \qquad (2.11)$$

where $\boldsymbol{c}(\boldsymbol{x}^*) = \Big(\boldsymbol{K}\big(\boldsymbol{x}^*, \boldsymbol{x}^{(1)}\big), \dots, \boldsymbol{K}\big(\boldsymbol{x}^*, \boldsymbol{x}^{(N)}\big)\Big)^{\boldsymbol{T}}$, $\boldsymbol{1}$ is a $p \times 1$ vector of ones, $\boldsymbol{r}(\boldsymbol{x}^*)$ is the vector of correlations between the point $\boldsymbol{x}^*$ and the N other sample points, $r_i(\boldsymbol{x}^*) = R\big(\boldsymbol{x}^*, \boldsymbol{x}^{(i)}\big)$ and $\boldsymbol{R}$ is a $N \times N$ correlation matrix between sample points, $R_{ij} = R\big(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\big)$.

With the correlation function and the moments of Y, we can estimate $\hat{y}(\boldsymbol{d}^*)$ and the mean square error $\hat{s}^2(\boldsymbol{x}^*)$.

Moreover, if $\boldsymbol{x}^*$ is at any exist sample point, that means $\boldsymbol{x}^* = \boldsymbol{x}^{(i)}$, then the mean square error turns to 0. In this situation, $\boldsymbol{r}\big(\boldsymbol{x}^{(i)}\big)$ is equal to $\boldsymbol{R}_{i\bullet}$ which signifies the ith column of the correlation matrix, the term $\boldsymbol{r}\big(\boldsymbol{x}^{(i)}\big)^T \boldsymbol{R}^{-1}$ is equal to $\boldsymbol{e}_i = (0, \dots, 0, 1, 0, \dots, 0)$ in which the ith element is 1. So that

$$\hat{y}\left(\boldsymbol{x}^{(i)}\right) = \mu_Y + \left(y\left(\boldsymbol{x}^{(i)}\right) - \mu_Y\right) = y\left(\boldsymbol{x}^{(i)}\right)$$
$$\hat{s}^2\left(\boldsymbol{x}^{(i)}\right) = {\sigma_Y}^2\left(1 - \boldsymbol{e}_i \boldsymbol{r}\left(\boldsymbol{x}^{(i)}\right)\right) = {\sigma_Y}^2(1 - R_{ii}) = 0 \tag{2.12}$$

according to Equation (2.7).

1.2.3 Ordinary Kriging (OK)

Here in this type, the average trend is constant but unknown, so that the Kriging model can be written as:

$$\hat{y}(\boldsymbol{x}^*) = \mu_Y + Z(\boldsymbol{x}) \tag{2.13}$$

With the condition of unbiased approximation mentioned before in Equation (2.10), we can deduce that:

$$\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*)\mu_Y - \mu_Y = 0 \Leftrightarrow \sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) = 1 \tag{2.14}$$

As our purpose is to minimize Equation (2.9) with the equality constraint Equation (2.10), so that we introduce a Lagrange multiplier $v$and let the partial derivative of λ_i equals to 0. The problem proposed by Equation (2.9) and (2.14) can be reformed as:

$$\mathcal{L}_{\boldsymbol{d}^*}(\lambda, \nu) = \sum_{i=1}^{N}\sum_{j=1}^{N} \lambda_i(\boldsymbol{x}^*)\,\lambda_j(\boldsymbol{x}^*) K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right) + Var\left(Y(\boldsymbol{x}^*)\right)$$
$$-2\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^*\right) + 2\nu\left(\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) - 1\right) \tag{2.15}$$

where the last term in right hand side equals to 0. And to cancel the partial derivative with respect to each component λ_i easily, we multiply by 2 before deriving. The following equations can be obtained after derivation:

$$\begin{cases} 2\sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right) - 2K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^*\right) + 2\nu = 0 \\ \sum_{i=1}^{N} \lambda_i(\boldsymbol{x}^*) = 1 \end{cases} \tag{2.16}$$

For an estimate of the function at the point $\boldsymbol{x}^*$, the weight values λ_i can be obtained by solving the problem reformulated by Equation (2.16):

$$\begin{bmatrix} K\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(1)}\right) & \cdots & K\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(N)}\right) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ K\left(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(1)}\right) & \cdots & K\left(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(N)}\right) & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1(\boldsymbol{x}^*) \\ \vdots \\ \lambda_N(\boldsymbol{x}^*) \\ \nu \end{bmatrix} = \begin{bmatrix} K\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^*\right) \\ \vdots \\ K\left(\boldsymbol{x}^{(N)}, \boldsymbol{x}^*\right) \\ 1 \end{bmatrix} \tag{2.17}$$

K can be calculated from the chosen correlation function in Table 2.1 and after calculation of λ_i, Equation (2.8) could be used to obtain the value of estimation.

1.2.4 Universal Kriging (UK)

In the universal Kriging model,

$$\hat{y}(\boldsymbol{x}^*) = \sum_{i=1}^{q} \beta_i \, t_i(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{2.18}$$

For the simplicity, the former equation can be written in matrix form as:

$$\widehat{\boldsymbol{y}} = \boldsymbol{T}\boldsymbol{b} + \boldsymbol{Z} \tag{2.19}$$

where $\boldsymbol{b} = \left[\beta_1, \dots, \beta_p\right]^T$, $\boldsymbol{T}$ is a $N \times q$ matrix whose ij th element is $t_j\left(\boldsymbol{x}^{(i)}\right)$ and $\boldsymbol{Z} = \left[Z\left(\boldsymbol{x}^{(1)}\right), \dots, Z\left(\boldsymbol{x}^{(N)}\right)\right]^T$ satisfies $\boldsymbol{Z} \sim \mathcal{N}(0, \boldsymbol{C})$, $\boldsymbol{C}$ is the matrix of covariance as shown in Equation (2.2). For the positive definite matrix $\boldsymbol{C}^{-1}$, there must exist a matrix $\boldsymbol{B}$ that satisfy $\boldsymbol{C}^{-1} = \boldsymbol{B}^T\boldsymbol{B}$. We modify Equation (2.19) by multiplying it by $\boldsymbol{B}$ for both sides:

$$\boldsymbol{B}\widehat{\boldsymbol{y}} = \boldsymbol{B}\boldsymbol{T}\boldsymbol{b} + \boldsymbol{B}\boldsymbol{Z} \tag{2.20}$$

With this model, the term $\boldsymbol{B}\boldsymbol{Z}$ is uncorrelated because that:

$$Cov(\boldsymbol{B}\boldsymbol{Z}) = \boldsymbol{B}Cov(\boldsymbol{Z})\boldsymbol{B}^T = \boldsymbol{B}\boldsymbol{B}^{-1}(\boldsymbol{B}^{-1})^T\boldsymbol{B}^T = \boldsymbol{I} \tag{2.21}$$

This is proved by using the fact:

$$Cov(\boldsymbol{Z}) = \boldsymbol{C} = \boldsymbol{B}^{-1}(\boldsymbol{B}^{-1})^{\boldsymbol{T}} \tag{2.22}$$

Thus the model in Equation (2.20) is consistent with the Gauss-Markov theorem. So finally with the generalized least squares method (Hansen, 2007), the coefficients are estimated by:

$$\widehat{\boldsymbol{b}} = ((\boldsymbol{B}\boldsymbol{T})^T\boldsymbol{B}\boldsymbol{T})^{-1}(\boldsymbol{B}\boldsymbol{T})^T\boldsymbol{B}\boldsymbol{Z} = (\boldsymbol{T}^T\boldsymbol{C}\boldsymbol{T})^{-1}\boldsymbol{T}^T\boldsymbol{C}^{-\mathbf{1}}\boldsymbol{Z} = \frac{\boldsymbol{T}^T\boldsymbol{R}^{-\mathbf{1}}\boldsymbol{Z}}{\boldsymbol{T}^T\boldsymbol{R}^{-\mathbf{1}}\boldsymbol{T}} \tag{2.23}$$

Other unknown parameters $\hat{s}^2$ and $\boldsymbol{\theta}$ can be estimated by the maximum likelihood method, the likelihood function is the probability density of the observations:

$$L(\boldsymbol{\theta}, \hat{s}^2|\widehat{\boldsymbol{y}}) = P(\widehat{\boldsymbol{y}}|\boldsymbol{\theta}, \hat{s}^2) = \frac{1}{(2\pi)^{\boldsymbol{n}/\mathbf{2}}|\boldsymbol{C}|^{\mathbf{1}/\mathbf{2}}} exp\left(-\frac{(\widehat{\boldsymbol{y}}-\mathbf{1}\mu)^T\boldsymbol{C}^{-1}(\widehat{\boldsymbol{y}}-\mathbf{1}\mu)}{2}\right) \tag{2.24}$$

$$\ln L(\theta, \hat{s}^2|\widehat{\boldsymbol{y}}) = -\frac{n}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{C}| - \frac{1}{2}(\widehat{\boldsymbol{y}} - \mathbf{1}\mu)^T\boldsymbol{C}^{-1}(\widehat{\boldsymbol{y}} - \mathbf{1}\mu) \tag{2.25}$$

where $|\boldsymbol{C}|$ is the determinant of the covariance matrix.

Then the estimator of variance $\hat{s}^2$ is:

$$\hat{s}^2 = \frac{1}{n}(\widehat{\boldsymbol{y}} - \mathbf{1}\mu)^T\,\boldsymbol{R}^{-1}(\widehat{\boldsymbol{y}} - \mathbf{1}\mu) \tag{2.26}$$

Substituting $\hat{s}^2$ on Equation (2.25):

$$2\ln L(\boldsymbol{\theta}|\widehat{\boldsymbol{y}}) = -n\ln(2\pi) - n\ln\hat{s}^2 - \ln|\boldsymbol{R}| - n \tag{2.27}$$

where the parameter $\boldsymbol{\theta}$ is estimated by maximizing the equation above subject to the constraint that all the elements of $\boldsymbol{\theta}$ should be positive.

The peculiarity of the kriging model is that the prediction error at any point can be also provided. This error is accessible from the estimation of the variance of the stochastic process.

In order to implement the Kriging meta-model of the objective function and also the constraints in design optimization problem with heavy models, a Matlab toolbox called DACE is used. The manual (Lophaven, 2002) and (Sacks et al., 1989) explain how to obtain the parameters using the toolbox in detail.

1.3 Meta-model based design optimization

In general, almost all the meta-model based design optimization strategies can be classified in three types:

- Sequential strategy
- Adaptive strategy
- Criterion-based strategy

For the first type, the meta-model building part and the optimization part are separated and the meta-model is not modified during the optimization process. The process of this strategy is shown in the Fig 2.1. A global meta-model is built by using the evaluation information on chosen samples and an optional validation step is made after the building. Once the meta-model is created, no more change will be made and it is used to substitute the expensive black-box model during the optimization process. The main advantage of this strategy is that it is simple to apply as the meta-model building part is separated to the optimization part. However, as there is no feedback, the accuracy depends on the initial samples, so that the optimization process may fail to find the global optimum solution if the choice of initial samples is not good or the number of sample points is not enough.

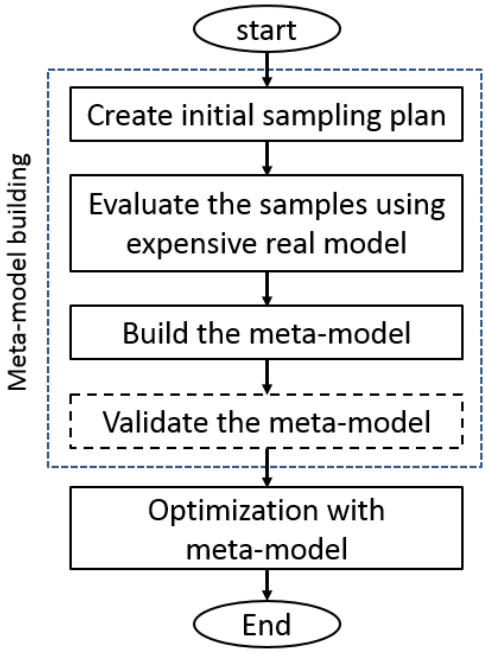


Fig. 2.1. Sequential strategy process

The second strategy is an adaptive approach: there is a feedback from optimization part to meta-model building part in order to adapt the approximation. First, the initial meta-model is built as in the sequential strategy, then once the optimum is found after optimization, this optimum is added to the samples and expensive model is run at this point then the meta-model is rebuilt. This loop will continue until the stop criterion is satisfied. The new meta-model integrate more information given by the optimum of each iteration so that the accuracy can be improved iteration by iteration. However, this strategy can only improve the local accuracy, in fact it has high dependence on the success of finding the global optimum by the first iteration, if a local optimum is found at first, then all the following added points may locate around this local optimum and it fails to converge. When many design variables are considered or the fine model is highly nonlinear, obtaining high global accuracy meta-model with a limited budget of fine model evaluations is not possible. The process of this strategy is in the Fig 2.2.

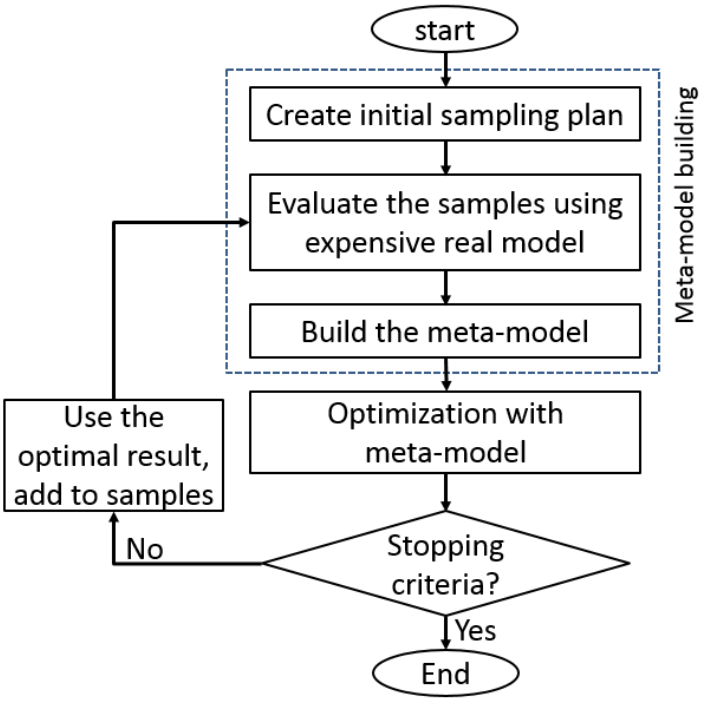


Fig. 2.2. Adaptive strategy process

The last strategy is called adaptive criterion-based strategy: this type of strategy is specially proposed to solve the problems that are highly nonlinear or has many variables. As explained before in the second strategy, it is difficult to achieve a global accuracy for this type of problem. In order to reduce the number of evaluations, this approach combines the exploration with exploitations by introducing an Infill Searching Criterion (ISC). Its difference with the second strategy is that this method does not seek to reach a high global accuracy of the meta-model. Instead, it will consider a balance between searching the optimum and improving the accuracy. All the solutions found by ISC in each iteration is a best combination of searching the optimal solution and exploration of the design space, then the optimization process will be guided by the ISC. This strategy is the

most complex but also the most efficient as it can improve the solutions with a reduced number of evaluations with the fine model. However owing to the complexity, as the rebuilding of meta-model is nested with ISC, if the searching analysis fails, the designer needs to adjust the simulation models parameters and resume the optimization process manually. Also, most of the ISC adds only one solution into the samples at each iteration, this may make the process to take lots of iterations. The process of this strategy is shown in Fig 2.3.

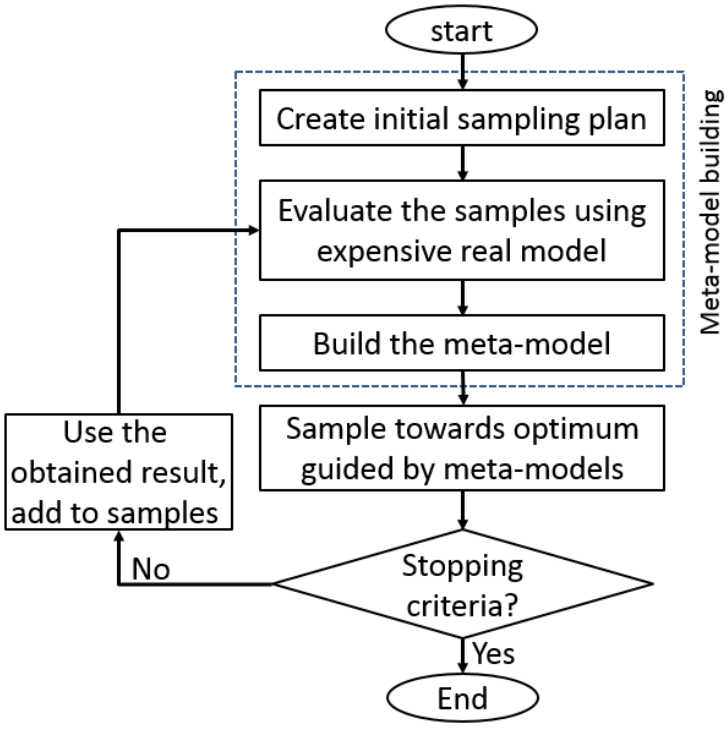


Fig. 2.3. Adaptive criterion-based strategy process

This strategy is chosen due to its adaptation with highly nonlinear and expensive fine models. The most common used method in this type of strategy is Efficient Global Optimization (EGO) and it will be presented in next section.

1.4 Efficient Global Optimization (EGO)

The EGO proposed by (Jones, Schonlau, & Welch, 1998) uses a kriging meta-model to approximate the real function from a set of observations, and it adds new points sequentially by maximize the Expected Improvement (EI) of the function which is considered as the infill criterion. The process of EGO is shown in the following figure. After the meta-model is built, EGO adds one point to the existing samples set by maximizing the value of EI, then the parameters of covariance functions are re-estimated and the Kriging model is updated. This process continues until a stopping criterion is satisfied. Usually the stopping criterion is that the maximum value of EI is small enough. (Chaudhuri & Haftka,

2013) discusses the stopping criteria used in surrogate model based optimization. In this manuscript, we use a fixed small positive value to stop the algorithm.

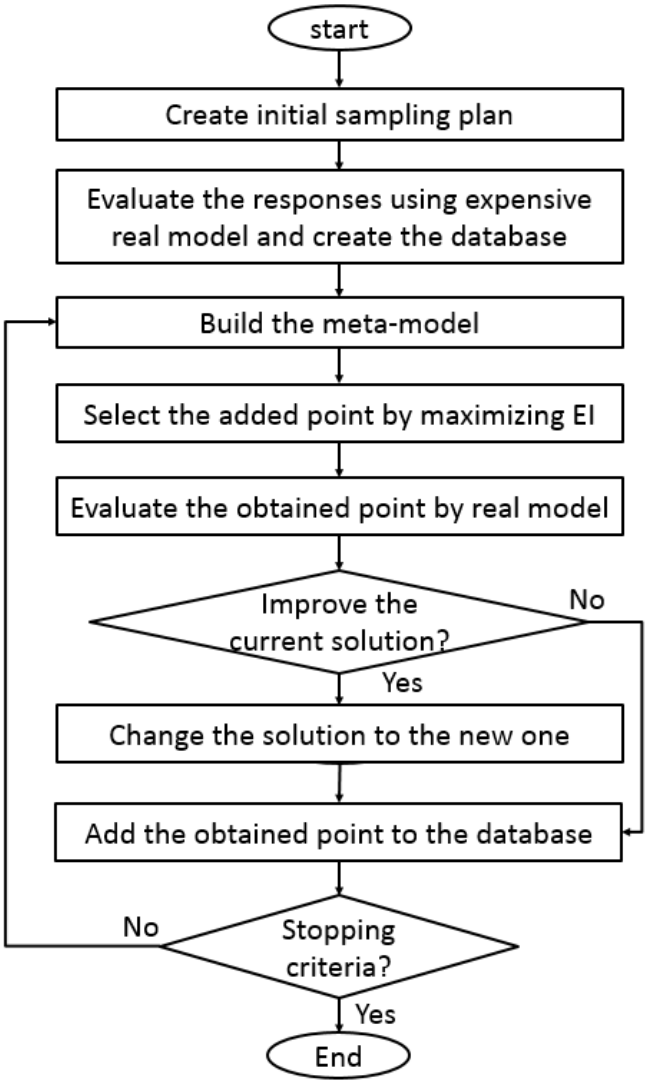


Fig. 2.4. The process of EGO

EGO iteratively creates a design point with the criterion EI which is particularly popular due to its advantages. First, it has the capability of exploration and exploitation at the same time, and it could lead the optimization to the global optimum effectively (Locatelli, 1997). Second, it does not have any parameters that needs to be adjusted. And last, with EI there is no need to choose another stopping criterion as it can itself be used as a stopping criterion: if the maximum value of EI is small enough, the optimization could be terminated (Forrester & Jones, 2008).

Expected Improvement indicates the amount of improvement we could expect if the function is evaluated at the point. So, searching the maximum EI can help us to find the point that has the most probability of improving the current solution.

We define the initial design sample set as $\boldsymbol{X} = \left[\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\right]^T$, and the evaluation function set at $\boldsymbol{X}$ is $\boldsymbol{Y} = \left[y\left(\boldsymbol{x}^{(1)}\right), \dots, y\left(\boldsymbol{x}^{(N)}\right)\right]^T$, then the function value $y(\boldsymbol{x})$ at some other point $\boldsymbol{x}$ is treated as the realization of a normally distributed random variable with mean $\hat{\boldsymbol{y}}(\boldsymbol{x})$ and standard deviation $\hat{s}(\boldsymbol{x})$ given by the kriging predictor built by information of $(\boldsymbol{X}, \boldsymbol{Y})$. Suppose $y_{min} = \min \boldsymbol{Y}$, the current best objective function value of all samples, the improvement is defined as:

$$I = \max\{0, y_{min} - \hat{y}(\boldsymbol{x})\} \tag{2.28}$$

The Expected Improvement is then calculated by:

$$EI(\boldsymbol{x}) = \begin{cases} [y_{min} - \hat{y}(\boldsymbol{x})]\Phi\left(z(\boldsymbol{x})\right) + \hat{s}(\boldsymbol{x})\phi\left(z(\boldsymbol{x})\right) & \text{if } \hat{s}(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}(\boldsymbol{x}) = 0 \end{cases} \tag{2.29}$$

where $z(\boldsymbol{x}) = \left(y_{min} - \hat{y}(\boldsymbol{x})\right)/\hat{s}(\boldsymbol{x})$. Here Φ and ϕ denote the cumulative distribution function (CDF) and probability density function (PDF) of standard normal distribution respectively. The design point that has the maximum value of EI will be taken as the next infill sample:

$$\boldsymbol{x}^{(N+1)} = \operatorname{argmax} EI(\boldsymbol{x}) \tag{2.30}$$

From the Equation (2.27) we could see that EI is a non-negative function and the two terms seek for exploitation and exploration respectively. And the EI value will augment in two conditions:

- $\hat{y}(\boldsymbol{x})$ is smaller than y_{min} thus the first term value increases
- $\hat{s}(\boldsymbol{x})$ has a large value so that the second term augment

It means that the first term aims to find a possible smaller value at the local position, while the second term contributes in increase the accuracy and global searching.

Besides the mentioned advantages of EI, there is a drawback because designer could not control the respective weights of exploration and exploitation. In order to tackle this difficulty, many ISC based on EI are proposed. These modified criteria will be presented in next section.

1.5 Infill Searching Criteria (ISC)

(Schonlau, 1997) introduced a Generalized Expected Improvement (GEI) criterion. It adds an additional non-negative integer parameter g in the improvement definition:

$$I^g = \max\left\{0, \left(y_{min} - \hat{y}(\boldsymbol{x})\right)^g\right\} \tag{2.31}$$

Therefore, the expression of GEI changes to:

$$EI^g(\boldsymbol{x}) = \hat{s}^g(\boldsymbol{x}) \sum_{i=0}^{g} (-1)^i \frac{g!}{i!(g-i)!} z(\boldsymbol{d})^{g-i} T_i \tag{2.32}$$

where $T_i = -\phi\big(z(\boldsymbol{d})\big) z(\boldsymbol{d})^{i-1} + (i-1)T_{i-2}$, starting with $T_0 = \Phi\big(z(\boldsymbol{d})\big)$ and $T_1 = -\phi\big(z(\boldsymbol{d})\big)$.

With the increasing of g, the method moves from local exploitation to global exploration. In particularly if $g = 1$ the criterion turns out to be the initial EI. The impact of the parameter g is shown in the Fig 2.5. In the figures, the solid line above is the real function while the dotted one is the meta-model, the curve below is the value of GEI. For $g = 1$ the GEI function rises significantly in two parts, left part has high model uncertainty while the right region has a higher probability of finding a better solution. Then if g grows larger, the left part value becomes greater than the right part, thus the contribution of second term in Equation (2.29) becomes more and more significant.

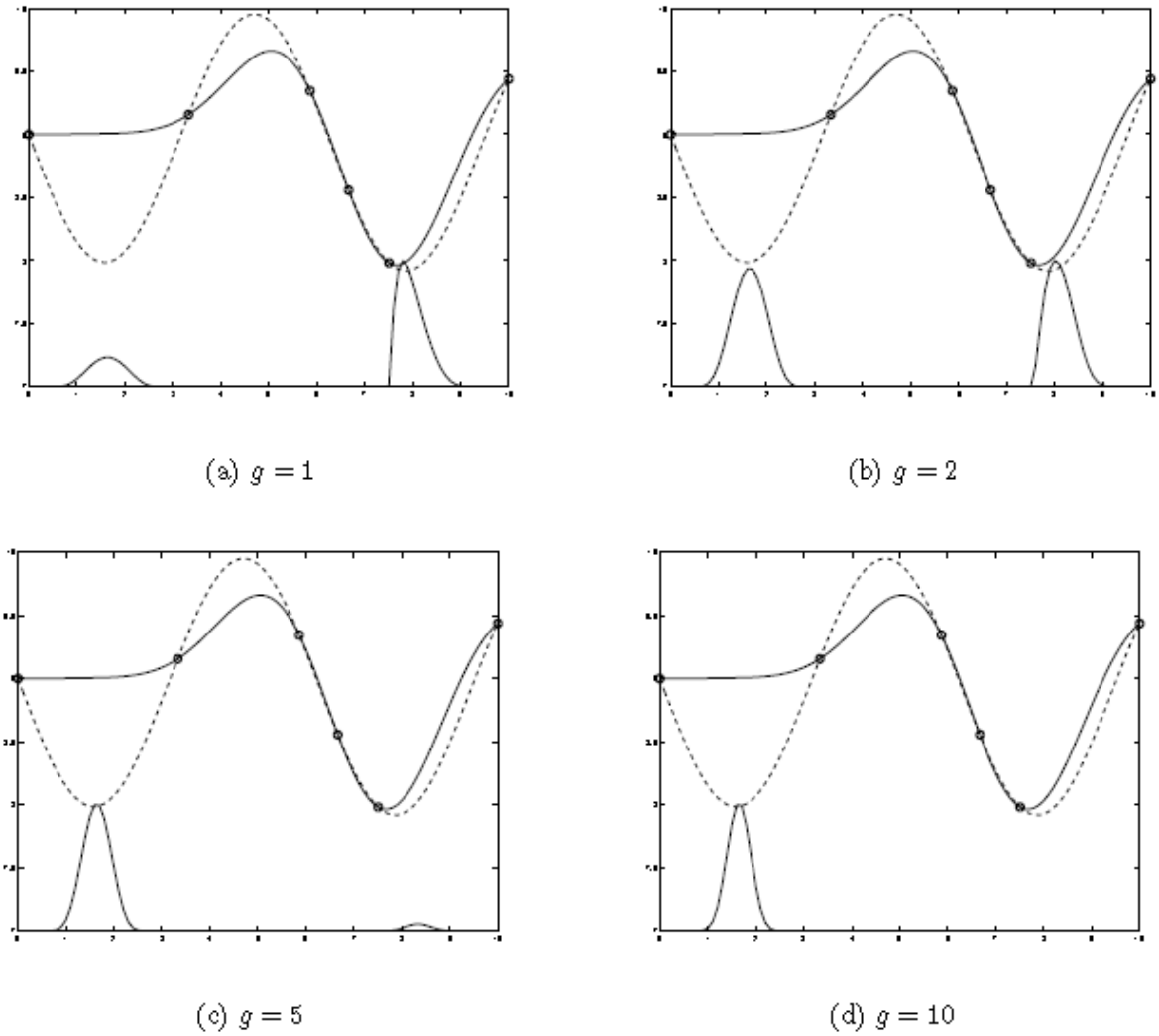


Fig. 2.5. The impact of different g values in GEI (Sasena, 2002)

(Sasena, 2002) introduced two varietal ISC, the first one aims to locate points that maximize the probability of being no greater than y_{min}, which is computed as:

$$WB_1(\boldsymbol{x}) = \Phi\left(\frac{(y_{min} - \hat{y}(\boldsymbol{x}))}{\hat{s}(\boldsymbol{x})}\right) \tag{2.33}$$

Notice that this formulation is exactly the same as that $g = 0$ for GEI. Therefore, this ISC focuses extremely on local searching. In fact it has the same idea as Probability Improvement (PI) (Jones, 2001). PI method focuses on the probability that the value $y(\boldsymbol{x})$ is smaller than a target value Y_T. It seeks for the point that has the highest probability that the function is less than the target Y_T:

$$P(y(\boldsymbol{x}) \leq Y_T) = F_y(T) = \int_{-\infty}^{T} f_y(t)dt \tag{2.34}$$

where F_y and f_y denote the CDF and PDF of y respectively. So the probability of improvement can be calculated as:

$$PI(\boldsymbol{x}) = P(y(\boldsymbol{x}) \leq Y_T) = \int_{-\infty}^{T} \frac{1}{s(\boldsymbol{x})\sqrt{2\pi}} e^{\frac{-(t-\hat{y}(\boldsymbol{x}))^2}{2\hat{s}^2(\boldsymbol{x})}} dt \tag{2.35}$$

with $u(\boldsymbol{x}) = Y_T - \hat{y}(\boldsymbol{x})/\hat{s}(\boldsymbol{x})$, then

$$PI(\boldsymbol{x}) = \int_{-\infty}^{\frac{Y_T-\hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}} \frac{1}{\sqrt{2\pi}} e^{\frac{-u^2}{2}} du = [\Phi(u)]_{-\infty}^{\frac{Y_T-\hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}} = \Phi\left(\frac{Y_T - \hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}\right) \tag{2.36}$$

The target Y_T is chosen by the designer and we can find that if $Y_T = y_{min}$, the expression of PI is the same as WB_1.

The other ISC introduced by (Sasena, 2002) attempts to minimize the expected value of the smallest observation once the infill samples have been added:

$$WB_2(\boldsymbol{x}) = \begin{cases} \hat{y}(\boldsymbol{x}) + [y_{min} - \hat{y}(\boldsymbol{x})]\Phi\big(z(\boldsymbol{x})\big) + \hat{s}(\boldsymbol{x})\phi\big(z(\boldsymbol{x})\big) & \text{if } \hat{s}(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}(\boldsymbol{x}) = 0 \end{cases} \tag{2.37}$$

It is very similar to the EI function, the only difference is that WB_2 has an additional first term that is the predicted value at the interest point. The advantage of this criterion is that it gives more weight to local search than EI. Also for EI function, it needs an initial sample point in the security domain to calculate y_{min} if there are constraints in the optimization. So a new problem arises if the constraints are complex and the security domain is small and/or discrete, there may be no point satisfying the constraints in the initial sample set, thus the EI function may fail to start as it always gives 0. WB_2 can avoid this problem as it adds an extra term and avoids the values return to 0.

(Sóbester, Leary, & Keane, 2005) proposed a Weighted Expected Improvement (WEI) criterion that can control the balance between local and global search by changing the value of the weight factor from 0 to 1. In order to distinguish with the weight parameter in formulations, here in ISC it is noted as $\alpha \in [0,1]$:

$$WEI(\boldsymbol{x}) = \begin{cases} \alpha[y_{min} - \hat{y}(\boldsymbol{x})]\Phi\big(z(\boldsymbol{x})\big) + (1-\alpha)\hat{s}(\boldsymbol{x})\phi\big(z(\boldsymbol{x})\big) & \text{if } \hat{s}(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}(\boldsymbol{x}) = 0 \end{cases} \tag{2.38}$$

A variant of the WEI was proposed by (Xiao, Rotaru, & Sykulski, 2012, 2013). It is called Adaptive Weighted Expected Improvement (AWEI) and uses two independent weight factors α_1 and α_2:

$$AWEI(\boldsymbol{x}) = \begin{cases} \alpha_1[y_{min} - \hat{y}(\boldsymbol{x})]\Phi\big(z(\boldsymbol{x})\big) + \alpha_2\hat{s}(\boldsymbol{x})\phi\big(z(\boldsymbol{x})\big) & \text{if } \hat{s}(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}(\boldsymbol{x}) = 0 \end{cases} \quad (2.39)$$

The factors can be calculated automatically at each iteration using a system of numerical rewards. Two potential rewards are calculated based on the average value of the mean square error (MSE) of all predicted points, representing the potential amount of rewards resulting from each of the two possible actions, exploration and exploitation, respectively. The determined rewards are then used to update the values of the two weighting coefficients. However, this approach is difficult to implement as it requires more parameters to be tuned.

(Lizotte, Greiner, & Schuurmans, 2012) introduced a modified EI by adding another parameter $\xi \geq 0$:

$$EI_{\xi}(\boldsymbol{x}) = \begin{cases} [y_{min} - \xi - \hat{y}(\boldsymbol{x})]\Phi\big(z(\boldsymbol{x})\big) + \hat{s}(\boldsymbol{x})\phi\big(z(\boldsymbol{x})\big) & \text{if } \hat{s}(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}(\boldsymbol{x}) = 0 \end{cases} \quad (2.40)$$

The principle of this criterion is the same as WB_2 : with higher values of the parameters ξ, the search pays more attention to exploration.

All the mentioned ISC above are adapted for objective function thus they can only solve unconstrained optimization problems which rarely occurs in real applications (Sasena, Papalambros, & Goovaerts, 2001). (Schonlau, 1997) analyses numbers of constraint handling techniques with applications and in this manuscript the most representative ones will be presented.

The first type is penalty function: a large negative constant is added to the expected improvement as a penalty in order to restrain it from choosing points inside infeasible domain (Sasena, Papalambros, & Goovaerts, 2000; Sasena, 2002).

Considering EI as example, in order to account for constraints, the formula should change like:

$$EIP(\boldsymbol{x}) = \begin{cases} EI(\boldsymbol{x}) + Penalty & \text{if } g(\boldsymbol{x}) > 0 \\ EI(\boldsymbol{x}) & \text{if } g(\boldsymbol{x}) \leq 0 \end{cases} \quad (2.41)$$

However, this approach has a main drawback that it is difficult to choose an appropriate value of the penalty.

(Schonlau, 1997) proposed a probability method that consists in multiplying the value of EI by the probability that the point is feasible, so that the magnitude of EI will reduce and if the point has a very low likelihood of feasibility the value will tend toward 0:

$$PF_i(\boldsymbol{x}) = P(g_i(\boldsymbol{x}) \leq 0) = \Phi\left(-\frac{\hat{g}_i(x)}{\hat{s}_{g_i}(x)}\right) \tag{2.42}$$

where $\hat{g}_i(\boldsymbol{x})$ is the predictor of the ith constraint g_i at the point $\boldsymbol{x}$ and $\hat{s}_{g_i}(\boldsymbol{x})$ is the standard deviation of predictor $\hat{g}_i$. Then the product is:

$$EI(\boldsymbol{x}) \times PF(\boldsymbol{x}) = EI(\boldsymbol{x}) \prod_{i=1}^{m} PF_i(\boldsymbol{x}) \tag{2.43}$$

m is the number of constraints.

There is a disadvantage for this approach because the influence of the probability on the criterion value is too strong so that if the optimum lies in the region around the limit-state, the algorithm may not be able to find it as its value is very small.

Another method which can handle constraints is called Expected violation (EV) proposed by (Audet, Denni, Moore, Booker, & Frank, 2000):

$$EV(\boldsymbol{x}) = \begin{cases} [\hat{g}(\boldsymbol{x}) - 0]\Phi\left(\frac{\hat{g}(x)-0}{\hat{s}_g(x)}\right) + \hat{s}_g(\boldsymbol{x})\phi\left(\frac{\hat{g}(x)-0}{\hat{s}_g(x)}\right) & \text{if } \hat{s}_g(\boldsymbol{x}) > 0 \\ 0 & \text{if } \hat{s}_g(\boldsymbol{x}) = 0 \end{cases} \tag{2.44}$$

Similar to EI, EV only changes the current minimum to the limit 0, the value of EV increases in the region where the constraints are likely to be non-violated or there is great uncertainty about the constraint models. If EV is less than a given threshold t_{EV}, the point can be considered as feasible. Then the new optimization problem is defined as:

$$\begin{aligned} &\max_{\boldsymbol{x}} EI(\boldsymbol{x}) \\ &s.t.\ EV(\boldsymbol{x}) \leq t_{EV} \end{aligned} \tag{2.45}$$

The fourth method is Constrained EI (CEI) proposed by (Sasena, Papalambros, & Goovaerts, 2001), it uses the Kriging meta-model of constraints directly thus the optimization problem turns to:

$$\begin{aligned} &\max_{\boldsymbol{x}} EI(\boldsymbol{x}) \\ &s.t.\ \hat{g}(\boldsymbol{x}) \leq 0 \end{aligned} \tag{2.46}$$

(Sasena, 2002) makes a comparison between EV, PF and CEI. Results show that CEI is the most accurate one.

Besides these approaches, (Bichon, Eldred, Swiler, Mahadevan, & McFarland, 2008) proposed another function called Expected Feasibility (EF) function to provide an indication of how well the true value of the response is expected to satisfy the constraints:

$$EF\big(\hat{g}(\boldsymbol{x})\big) = \int_{-\varepsilon}^{\varepsilon} [\varepsilon - |\hat{g}(\boldsymbol{x})|] f_{\hat{g}} dg \tag{2.47}$$

where $f_{\hat{g}}$ is the PDF of $\hat{g}(\boldsymbol{x})$ and ε is proportional to the standard deviation of the predictor $\big(\varepsilon \propto s_g\big)$. The former equation can be calculated as:

$$EF(\hat{g}(\boldsymbol{x})) = \hat{g}\left[2\Phi\left(-\frac{\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right) - \Phi\left(\frac{-\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right) - \Phi\left(\frac{\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right)\right]$$

$$-\hat{s}_g\left[2\phi\left(-\frac{\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right) - \phi\left(\frac{-\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right) - \phi\left(\frac{\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right)\right] + \varepsilon\left[\Phi\left(\frac{\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right) - \Phi\left(\frac{-\varepsilon-\hat{g}(\boldsymbol{x})}{\hat{s}_g(\boldsymbol{x})}\right)\right] \quad (2.48)$$

EF balances also between exploration and exploitation like EI and it is used in the method called Efficient Global Reliability Analysis (EGRA) proposed also by (Bichon et al., 2008). The main process of EGRA is similar to EGO and usually the value of ε is chosen as $2\hat{s}_g$.

The advantage of this methods is that it considers the constraints independently without the objective function, so it can obtain a very accurate meta-model for constraints. However, the main drawback is also attributed to this separation. As it only considers the meta-model of constraints, thus it divides the original optimization problem into several sub-problems and after each meta-model of constraints have been built, other methods or ISC are still needed to create meta-model of objective functions and solve the design optimization. The total process therefore becomes a tedious work.

2 Adaptive methods for optimization with uncertainty

As it can be seen from the state of the art, adaptive methods and infill searching criteria for deterministic design optimization problems are quite developed. Nevertheless for the robust or/and reliability design optimization problems, these techniques may not be completely suitable as the formulations changed. Hence, for considering the uncertainties, new strategies and ISC for adaptive methods should be studied. This part will detail the methods including the choice of the ISC and the positioning of sample enrichments in the optimization process for each aforementioned type of optimization with uncertainty.

2.1 Adaptive Kriging based WCO

There are much less authors who payed attention to WCO with adaptive criterion-based kriging method compared to other categories of optimization with uncertainty like RBDO. One of the most difficult question is how to predict the worst-case for every sample point. The worst-case optimization problem is usually referred to as a minimax problem, so there is an extra optimization sub-problem.

(Zhou & Zhang, 2010) proposed to build meta-models for the worst-case directly. That is, after initial sample set has been chosen $\boldsymbol{d}_{samp} = [\boldsymbol{d}_1, \dots, \boldsymbol{d}_n]$ where n denotes the number of samples and $\boldsymbol{d}$ is the mean of random variable $\boldsymbol{x}$. Traditional WCO or worst-vertex-based WCO method is used to evaluate the worst-case of each initial point. Response sets of worst-case objective and constraints values $\boldsymbol{F_w}$ and $\boldsymbol{G_w}$ are filled with the worst-case found around each sample, then meta-models of worst-case functions $\hat{f}_w$

and $\widehat{\boldsymbol{g}}_w$ are built to solve the minimization problem. The solution of optimization $\boldsymbol{d}_{new}$ will be added to the sample set and evaluated to rebuild the meta-models until convergence. Fig 2.6 presents the flowchart of this method.

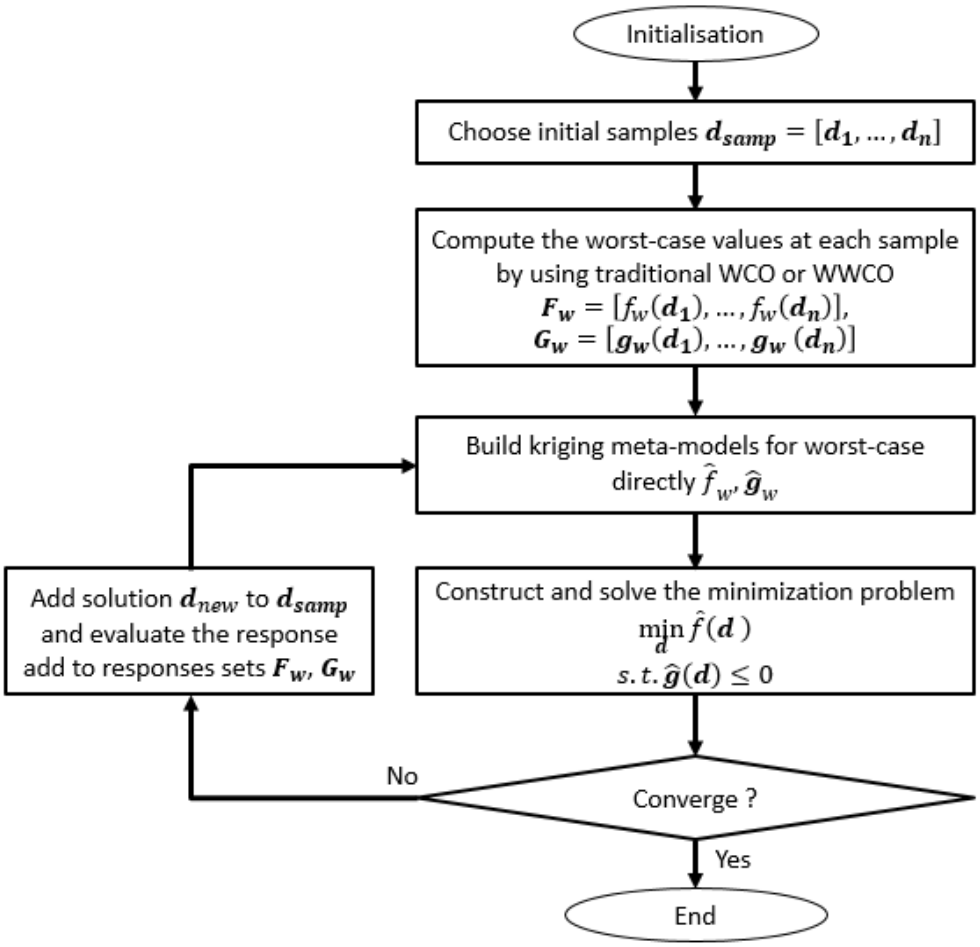


Fig. 2.6. The flowchart of the mentioned approach from (Zhou & Zhang, 2010)

However, for searching the worst-case value of each sample, more than one evaluation is needed. For example, if WWCO is used, then two bounds at the uncertainty set of each dimension need to be evaluated, thus $2p + 1$ evaluations for the current design point where p is the dimension of input variables. So n samples bring $n \times (2p + 1)$ evaluations.

Moreover, the worst-case value of the objective function or constraints at any given point depends not only on the information given by that point but also on the neighboring points. Thus, the nonlinearity and non-derivative of the worst-case solution is higher than initial objective or constraints due to its implicit form. If we build the meta-models directly for the worst-case solution, the precision may not be accurate enough. So that this approach takes more evaluations than expected and could not achieve a high precision as expected.

(Ur Rehman & Langelaar, 2017) investigated another approach which can handle the problems with constraints. This algorithm is called Efficient Global Robust Optimization

under Implementation Error (EGRO-IE). Although it has the name of robust, it uses worst-case optimization in essence. In the similar way to all other adaptive criterion-based meta-model methods, it starts with the construction of a meta-model based on initial samples and responses. However, the key points of this approach are: a reference optimum r on meta-model is used to replace the best value y_{min} in original EI, for calculating r, an optimization with meta-model is used:

$$\begin{aligned} r &= \min_{\boldsymbol{d}\in\mathbb{R}^p} \max_{\boldsymbol{\xi}\in U(\boldsymbol{d})} \hat{f}(\boldsymbol{d}+\boldsymbol{\xi}) \\ s.t.\ \max_{\boldsymbol{\xi}\in U(\boldsymbol{d})} &\hat{g}(\boldsymbol{d}+\boldsymbol{\xi}) + k\hat{s}_g(\boldsymbol{d}+\boldsymbol{\xi}) \le 0 \end{aligned} \tag{2.49}$$

where $U(\boldsymbol{d})$ is the uncertainty set around $\boldsymbol{d}$ defined as: $U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^p | \boldsymbol{d} - k\boldsymbol{\sigma} \le \boldsymbol{\xi} \le \boldsymbol{d} + k\boldsymbol{\sigma}\}$, $\boldsymbol{\sigma}$ is the standard deviation of uncertain variables with normal distribution. The parameter k is the confidence level as in Chapter 1, it is a measure of how conservative we want to be with respect to the meta-model prediction error, larger values indicate more conservative.

Then a modified EI and PF criterion is applied, and for each location at which this criterion needs to be computed, a worst-case Kriging prediction with respect to the uncertainty set is evaluated on the meta-model. The highest value for the worst-case Kriging prediction over the reference robust optimum is chosen as the new location of sample. The modified criterion EI multiplied by PF is calculated as follows:

$$\begin{aligned} EI_w = E[I_w(\boldsymbol{d})] &= \left(r - \hat{f}_{max}\right)\Phi\left(\tfrac{r-\hat{f}_{max}}{\hat{s}}\right) + s\phi\left(\tfrac{r-\hat{f}_{max}}{\hat{s}}\right) \\ PF_w = P[g_w(\boldsymbol{d}) \le \boldsymbol{0}] &= \Phi\left(-\tfrac{\hat{g}_{max}}{\hat{s}_g}\right) \end{aligned} \tag{2.50}$$

The worst-case Kriging predictions of objective function and constraints with respect to the uncertainties in design variable space are:

$$\hat{f}_{max}(\boldsymbol{d}) = \max_{\boldsymbol{\xi}\in U(\boldsymbol{d})} \hat{f}(\boldsymbol{d}+\boldsymbol{\xi}) \tag{2.51}$$

$$\hat{g}_{max}(\boldsymbol{d}) = \max_{\boldsymbol{\xi}\in \boldsymbol{U}(\boldsymbol{d})} \hat{g}(\boldsymbol{d}+\boldsymbol{\xi}) \tag{2.52}$$

The flowchart is in Fig 2.7. The advantage of this method is that we use meta-model to find the worst case for each design point so as to reduce the number of evaluations. However, it is not accurate enough as PF has a great influence when the searching point is close to the limit-states. In addition, the final solution is chosen as r, but this value is calculated by the meta-models, which means that this point has not been evaluated and the accuracy is further reduced.

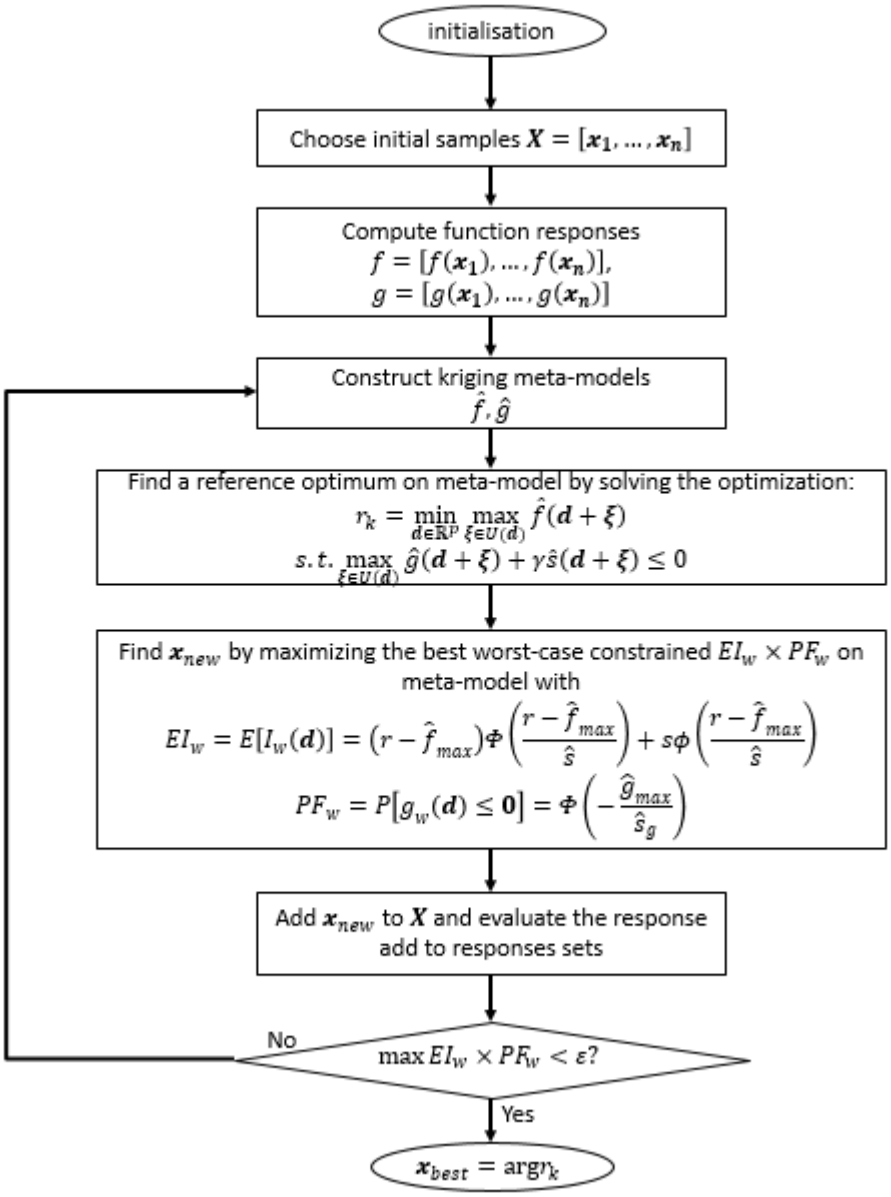


Fig. 2.7. The flowchart of worst-case optimization with constrained problems

2.1.1 New infill strategy for WCO

All the mentioned methods for WCO have their own drawbacks. Thus, a new strategy which can balance the speed of convergence and the precision is needed. In this manuscript two strategies are proposed.

First, as discussed above, meta-models built on worst-case objective function and constraints directly are not accurate enough due to the fact that they cannot consider all the information belonging to not only the current design point but also to the neighborhood. Thus, we decide to use meta-models based on original deterministic objective and constraints, then use them to predict the worst-case.

Besides, for ISC, (Li, Rotaru, & Sykulski, 2016) proposed a new criterion to add infill points at each iteration called the Worst-Case Expected Improvement (WCEI) for unconstrained problems. This expected improvement measure is recalculated from deterministic EI by taking the minimal value with the worst-case region of that design point:

$$WCEI(\boldsymbol{d}) = \max_{\boldsymbol{d}} \left\{ \min_{\xi} [EI(\boldsymbol{d} + \boldsymbol{\xi})], 0 \right\} \tag{2.53}$$

where $\boldsymbol{\xi}$ is a set of points located within the neighborhood region $\boldsymbol{U}(\boldsymbol{d})$ of the unknown point $\boldsymbol{d}$. With this ISC, new infill points are selected and evaluated at each iteration. After the stopping criterion is satisfied, the process finishes and exports the solution of the last iteration as the worst-case optimum. The stopping criterion is:

$$WCEI^k(\boldsymbol{d}) - WCEI^{k-1}(\boldsymbol{d}) < \varepsilon \tag{2.54}$$

where $WCEI^k(\boldsymbol{d})$ is the value of ISC in kth iteration and ε is a very small positive number decided by designer. Moreover, for the constrained problems, since criterion PF works not so good, we still consider $\hat{g}(\boldsymbol{d}_w) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \hat{g}(\boldsymbol{\xi})$ as constraints.

The second strategy proposed in this manuscript uses also the meta-models for original objective function and constraints as the former one. However, the ISC and the strategy for adding infill points are changed. After the meta-models of original objective and constraints are built with initial samples, a modified EI which can be adjusted to WWCO formulation with a little change:

$$EI_w(\boldsymbol{d}) = \left(\hat{f}_{w_{min}} - \hat{f}_w(\boldsymbol{d})\right) \Phi\left(\frac{\hat{f}_{w_{min}} - \hat{f}_w(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) + s(\boldsymbol{d}) \phi\left(\frac{\hat{f}_{w_{min}} - \hat{f}_w(\boldsymbol{d})}{\hat{s}(\boldsymbol{d})}\right) \tag{2.55}$$

where $\hat{f}_{w_{min}}$ is the current worst-case value given by WWCO using meta-model, $\hat{s}(\boldsymbol{d})$ is the MSE of meta-model $\hat{f}$ at design point $\boldsymbol{d}$. $\hat{f}_w$ is the prediction of worst-case of objective calculated by:

$$\hat{f}_w(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} \hat{f}(\boldsymbol{d} + \boldsymbol{\xi}) \tag{2.56}$$

However, when WWCO is used, only the vertex are taken into account, if the function changes greatly in the area $\boldsymbol{U}$, obviously this method is not sufficient to find the correct worst-case solution (Song, Li, Rotaru, & Sykulski, 2014). Thus, the gradient index GI is also introduced in our ISC for the purpose of finding a more credible solution:

$$GI(\boldsymbol{d}) = \max_{i=1,\ldots,p} \left(\frac{\partial \hat{f}}{\partial d_i}\right) \tag{2.57}$$

So that the final ISC will be the product of EI and GI.

The same example with 2 variables and 3 constraints in the former chapter will be used after to test these two approaches.

2.1.2 Mathematical example

The optimization problem of the example is presented in Equations (1.64) and (1.65) in Chapter 1 section 2. Following two figures present the results for above two methods. Colored contours are for deterministic objective function. Blue dotted lines and red dotted lines are deterministic constraints $g_i(\boldsymbol{d}) = 0$ $(i = 1,2,3)$ and worst-case constraints $g_{w,i}(\boldsymbol{d}) = \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} g_i(\boldsymbol{d} + \boldsymbol{\xi}) = 0$, respectively. Black solid lines and green solid lines are meta-models of deterministic constraints and worst-case constraints, respectively. Green points present 20 initial samples, red points are added during iterations. The reliability index β_t is considered as 2.

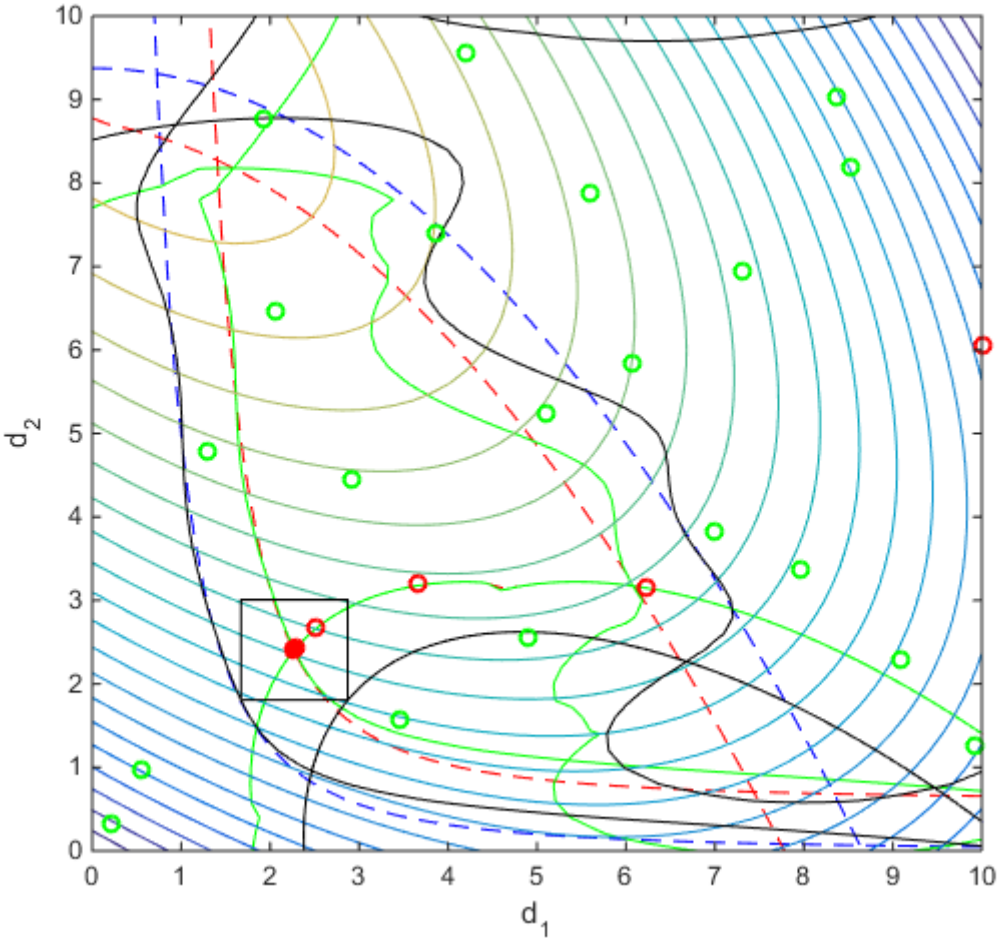


Fig. 2.8. The final iteration of meta-models of worst-case using WCEI

From the Fig 2.8, it can be seen that the meta-models of constraints seems not very accurate especially at the positions far from the optimum, it is easy to understand this phenomenon because we focus on searching the worst-case optimum and improving the accuracy of meta-model at the same time. The new infill points are located where the minimal EI around the target point is the largest, and the parts far from optimum are not very interesting. There are 8 iterations before convergence: except the three first points, the other 5 points added are located in almost the same location.

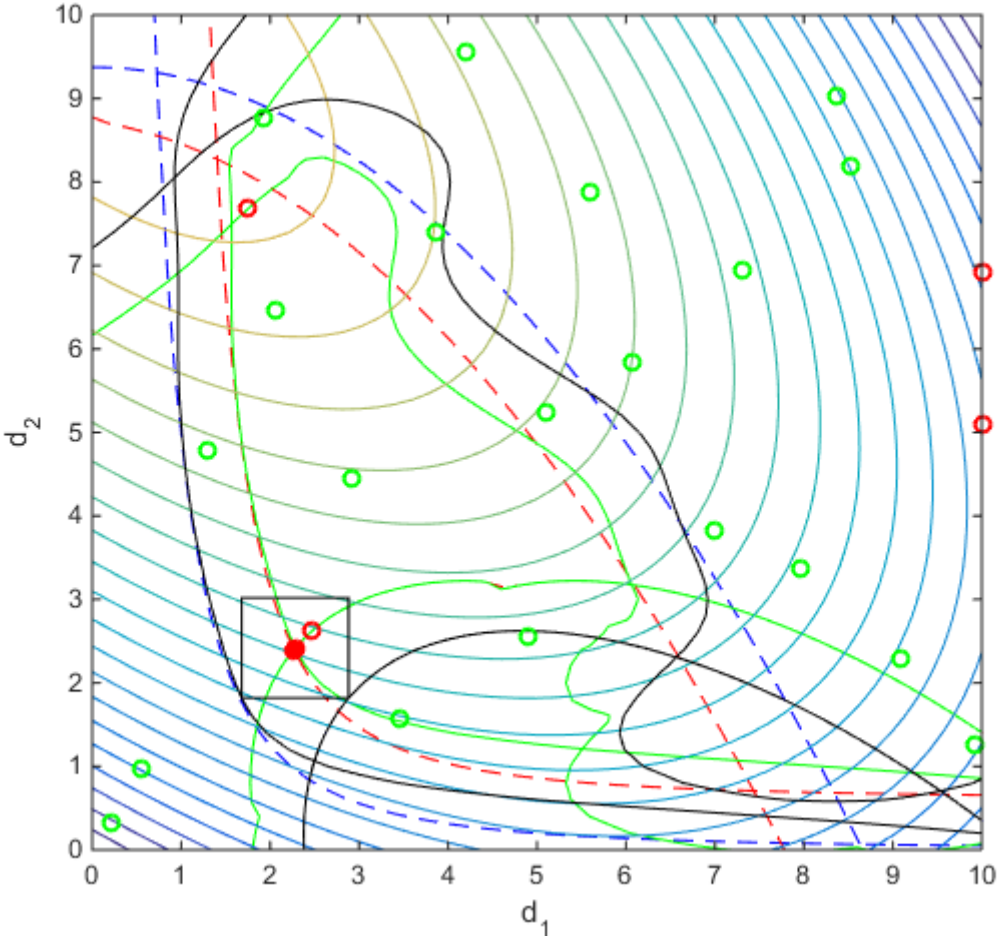


Fig. 2.9. The final iteration of meta-models of worst-case using EI and GI

Then from the Fig 2.9 and Table 2.2 it can be seen that worst-case using EI and GI method finds a worst-case optimum less accurate than the former one. After convergence, the meta-model and mathematic model of original objective function still have a visible difference. Moreover, in Table 2.2, results of WWCO using mathematical model without kriging are also given for comparing. It is obvious that the method using WCEI could find a more accurate solution than using $EI \times GI$ while both of them have almost the same number of evaluations.

Table 2.2. Results of mathematical example using different strategies of WCO

ISC	Number of iterations	Number of evaluations	Optimal solution
$EI \times GI$	7	20+7	[2.2585; 2.3904]
$WCEI$	8	20+8	[2.2700; 2.4051]
Mathematical model without kriging	--	121	[2.2664; 2.4006]

2.2 Adaptive Kriging based RDO

As RDO aims to find a minimal variance, the robust optimum no longer stays around the deterministic optimum, additionally a deterministic local optimum may become the robust global solution.

So the most straightforward but not efficient method is to consider separately the part of meta-model construction and the part of RDO. For example, (Lee & Kang, 2006; Lee & Park, 2006) proposed to build surrogate approximation models for initial objective function and constraints by evaluating some chosen samples. Then, a RDO problem is solved using these surrogate models with the mean and variance are computed by Monte-Carlo simulations.

With enough initial samples we can get a meta-model with high precision in all domain and we will not miss any local optima which are candidates of the robust solution. However, the shortcoming is that this method depends greatly on the initial samples as the kriging model is not adaptive. If the number of samples is too large, the computing time is excessive and if the number is too small, it may not obtain an accurate meta-model and lead to a wrong result.

In order to handle this drawback, adaptive criterion based kriging methods for RDO are proposed. Two types of methods are proposed to get RDO in tune with adaptive kriging: the first one adjust the ISC to the robust problem, and the other one is to build meta-models for robust objective and constraints.

For the first type, the roughest approach is using the classical EI directly on meta-model of original objective function. The only difference with (Lee & Park, 2006) is that this approach adds an ISC, so that it may need less sample points in total to get an accurate meta-model. However, the RDO problem is still separated into two parts, the first part using EGO to obtain a meta-model with enough precision, the second part use RDO for searching the robust optimum. Moreover, as the robust objective and constraints are very different with deterministic ones, it does not make any sense to treat random variables in the same way as deterministic variables in the evaluation of the expected improvement (Jurecka, 2007).

It seems that only adjusting the ISC to original problem with random variables for RDO could not satisfy our aim of mixing the meta-model construction and optimization together. Fortunately, we have another type of method, that is building a new meta-model for robust objective function and constraints so that ISC can work on it.

As the goal is to find a robust optimum that is insensitive to uncertainties in the design process, the robustness measure such as the standard deviation of performance is evaluated and minimized to achieve this goal directly (Doltsinis, Kang, & Cheng, 2005). However, as one main feature of meta-models is their smoothing properties, local sensitivities obtained from a global model are in general not reliable (Lönn, Jergeus, &

Nilsson, 2013). (David, Fyllingen, & Nilsson, 2008; Sun et al., 2011) presented an approach to robust optimization by using two meta-models: one is for the mean and other is for the standard deviation of the initial objective function (these articles consider only the problem without constraints). The authors use Artificial Neural Network (ANN) to approximate the mean and standard deviation of the responses over the design variables' space. Then, the formulation of robust optimization mentioned in the previous chapter can be used to find a solution. If convergence is achieved then the algorithm stops otherwise the meta-model of mean and standard deviation are rebuilt by choosing other samples to build new meta-models. The flowchart is shown in Fig 2.10.

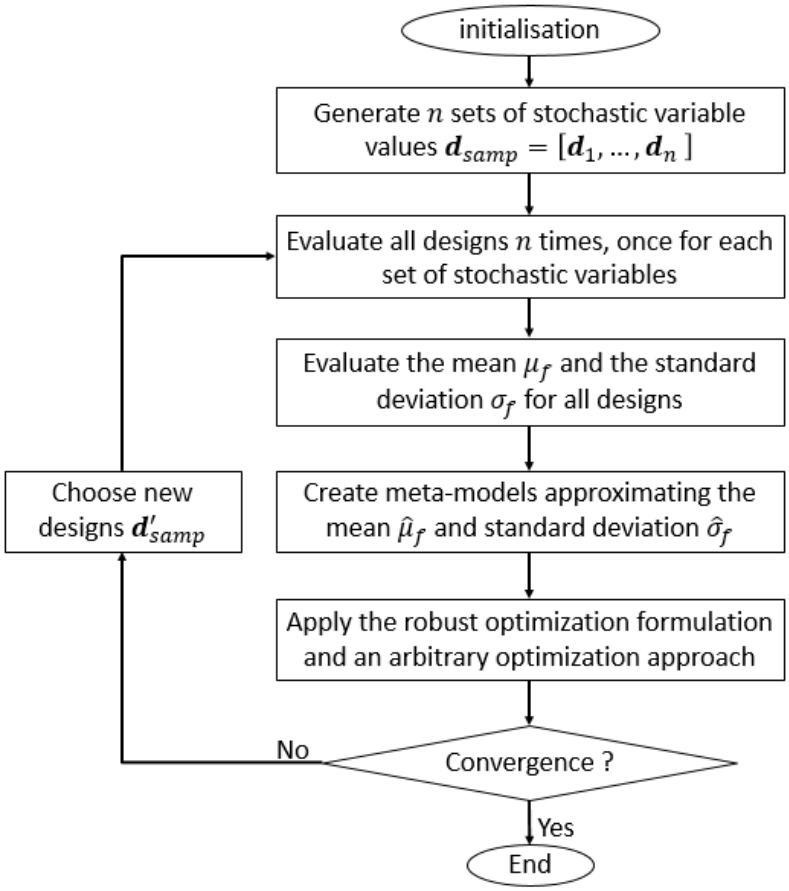


Fig. 2.10. Flowchart for the methodology used in (David, Fyllingen & Nilssona, 2008)

However, new design points are created at each iteration randomly and the meta-models are rebuilt each time without considering the previous designs. This wastes the previous evaluations and is time consuming.

In order to choose new designs more efficiently and associate the new infill points with previous samples, ISC should be used. (Shimoyama, Sato, Jeong, & Obayashi, 2012) proposed an ISC called Expected Hyper Volume Improvement (EHVI) which could be used for multi-objective optimizations. EHVI focuses on the improvement of the front of non-dominated solutions in the objective functions' space whose axis are the mean and standard deviation. Nevertheless, the new infill points are just forecasts of new global

optima may achieve from existing solutions and they are not a real solution found by calculation, thus, the so-called global optima may be not correct and lead the process to a wrong direction.

To accelerate the process and find a more suitable infill point, new strategy need to be proposed for RDO problem.

2.2.1 New infill strategy for RDO

In this chapter, a new method of RDO with adaptive criterion-based Kriging meta-model is proposed. As has been explained, several difficulties arise if we focus on using ISC with meta-models of original objective and constraints when random variables are taken into account.

In order to combine the meta-model building and robust optimization together, ISC should act on the robust objective and/or constraints directly. Thus, the goal is to find an infill point that is most promising with respect to the robustness criterion instead of focusing on the improvement on the surrogate models.

With this purpose, the means $\mu_f, \boldsymbol{\mu_g}$ and standard deviations $\sigma_f, \boldsymbol{\sigma_g}$ of original objective function and constraints are used as response surfaces and the meta-models are constructed for them straightforwardly. Then, robust objective function and constraints can be built from these surrogate moments.

The flowchart can be seen from Fig 2.11. First after some initial points are selected, the approaches presented in Chapter 1 like Monte Carlo simulation or Taylor based method can be used to calculate the moments of original objective function and constraints. For example, if we use the first order Taylor based method, the moments can be calculated as:

$$\begin{cases} \hat{\mu}_f = f(\boldsymbol{d}) \\ \hat{\sigma}_f^2 = \sum_{i=1}^{p} \left(\frac{\partial f(\boldsymbol{d})}{\partial d_i}\right)^2 \sigma_i^2 \end{cases} \tag{2.58}$$

$$\begin{cases} \hat{\mu}_{g_j} = g_j(\boldsymbol{d}) \\ \hat{\sigma}_{g_j}^2 = \sum_{i=1}^{p} \left(\frac{\partial g_j(\boldsymbol{d})}{\partial d_i}\right)^2 \sigma_i^2 \end{cases} \tag{2.59}$$

where p is the dimension of input parameter, d_i is the ith element of mean of input parameter, σ_i is the ith element of standard deviation of input parameter g_j is the jth constraint.

Then, the kriging meta-model of these moments are constructed in order to create the robust formulations. For example, Formulation 7 in Table 1.3 in previous chapter:

$$\begin{cases} \hat{f}_r(\boldsymbol{d}) = \omega \frac{\hat{\mu}_f(\boldsymbol{d})}{|\mu_{f_0}|} + (1-\omega)\frac{\hat{\sigma}_f(\boldsymbol{d})}{\sigma_{f_0}} \\ \widehat{\boldsymbol{g}}_r = \widehat{\boldsymbol{\mu}}_{\boldsymbol{g}}(\boldsymbol{d}) + k\widehat{\boldsymbol{\sigma}}_{\boldsymbol{g}}(\boldsymbol{d}) \end{cases} \tag{2.60}$$

where μ_{f_0} and σ_{f_0} can be chosen as the moments calculated with the initial samples.

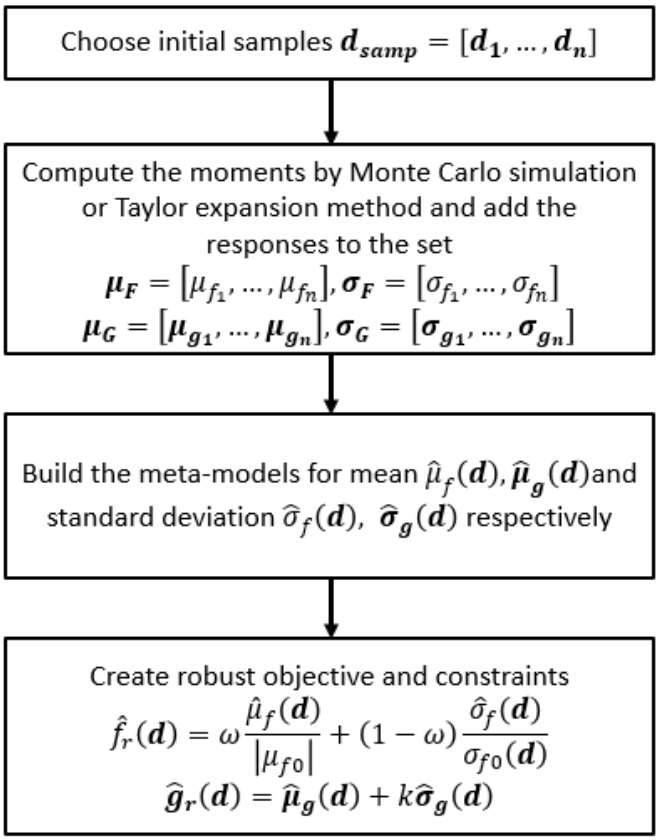


Fig. 2.11. Flowchart for constructing meta-models for moments and create robust formulation

This part is almost the same with the first iteration of the mentioned method proposed by (David et al., 2008), the only two differences are that we use uncertainty propagation to calculate the moments and use kriging to build meta-model.

Although there are two response surfaces: mean and standard deviation, they are not involved in the optimization problem directly. Only the robust objective function and constraints are used within the RDO problem. ISC can act on the problem straightly and ensures that the added sample points focus on finding the robust solution and increasing the accuracy of meta-model in the same time. So, there is no need to consider criteria like EHVI to forecast the next sample point by considering the Pareto front of mean and standard deviation.

With this strategy, a more suitable ISC is needed. We first consider only the objective function and a Modified Weighted EI is proposed. It is a variation of the Weighted EI (WEI) criterion proposed in (Sóbester et al., 2005). WEI seems to be more suitable as it adds weights into EI expression to balance exploration and intensification. Thus, we can pay more attention to searching the robust solution than increasing the accuracy for the meta-model by changing the weight.

$$\mathrm{W}EI = \begin{cases} \alpha\left(f_{min} - \hat{f}\right)\Phi(z) + (1-\alpha)s_f\phi(z) & \text{if } \hat{s}_f > 0 \\ 0 & \text{if } \hat{s}_f = 0 \end{cases} \tag{2.61}$$

where α is the weight factor, $z(\boldsymbol{d}) = \left(f_{min} - \hat{f}(\boldsymbol{d})\right)/\hat{s}(\boldsymbol{d})$. Small value of weight prevents WEI from converging to a local minimum. However, in constrained optimization, both EI and WEI require an initial sampling inside the security domain to start the improvement. This condition is quite difficult to satisfy for some complicated problems as their security domains may be small and sometimes discontinuous. To avoid this issue, a Modified WEI (MWEI) combined with the surrogate objective function is proposed:

$$\mathrm{MW}EI = \mathrm{W}EI - \alpha\hat{f} \tag{2.62}$$

In order to better understand the effects of ω, the same example as in (Xiao et al., 2013) is used and results for different values of weight are shown in Table 2.3. LM means local minimum and GM means global minimum. Thus, for searching the global optimum, the weight is typically taken equal to 0.1 as it provides to a global optimum with less iterations.

Table 2.3. Result of different value of weight

Value of weight	Number of iteration	Minimum found by meta-model	Exact value of the minimum
1	9	-302.5256	-300.5446 (LM)
0.8	9	-302.5248	-300.5446 (LM)
0.5	9	-302.5240	-300.5446 (LM)
0.2	22	-421.0057	-418.9827 (GM)
0.1	13	-420.9068	-418.9825 (GM)
0	fails		

Therefore, to adjust MWEI to RDO problem, $\hat{f}_r$ is used to replace $\hat{f}$:

$$\mathrm{MW}EI_{f_r} = \begin{cases} \alpha\left(\left(f_{r\,min} - \hat{f}_r\right)\Phi\left(z_{f_r}\right) - \hat{f}_r\right) + (1-\alpha)s_{f_r}\phi\left(z_{f_r}\right) & \text{if } \hat{s}_{f_r} > 0 \\ 0 & \text{if } \hat{s}_{f_r} = 0 \end{cases} \tag{2.63}$$

where $f_{r\,min}$ is the current minimal value of robust objective, s_{f_r} is the MSE of $\hat{f}_r$, $z_{f_r} = \left(f_{r\,min} - \hat{f}_r\right)/\hat{s}_{f_r}$.

For the constraints, as has been discussed in state of the art and adaptive kriging based WCO method, the meta-model of constraints themselves is more accurate, so in robust optimization, we also use robust constraints without making any changes. In the same way as objective function, mean $\boldsymbol{\mu_g}$ and standard deviation $\boldsymbol{\sigma_g}$ of constraints are taken as responses surfaces. Thus, the robust constraints with meta-model are written as:

$$\hat{\boldsymbol{g}}_{\boldsymbol{r}} = \hat{\boldsymbol{\mu}}_{\boldsymbol{g}} + k\hat{\boldsymbol{\sigma}}_{\boldsymbol{g}} \tag{2.64}$$

Then our purpose is to find the point which could maximize the value of MWEI with the robust constraint:

$$\begin{gathered} \max_{d} \mathrm{MW}EI_{f_r}(\boldsymbol{d}) \\ s.t.\, \hat{\boldsymbol{g}}_{\boldsymbol{r}}(\boldsymbol{d}) < 0 \\ \boldsymbol{d}^L + k\sigma \le \boldsymbol{d} \le \boldsymbol{d}^U - k\sigma \end{gathered} \tag{2.65}$$

where $\boldsymbol{d}^L$ and $\boldsymbol{d}^U$ denote the lower and upper bounds of $\boldsymbol{d}$, k is the confidence level.

The process of this strategy is shown in the following figure. Then, the same mathematic example with two variables and three constrains will be used here to assess the efficiency of this method.

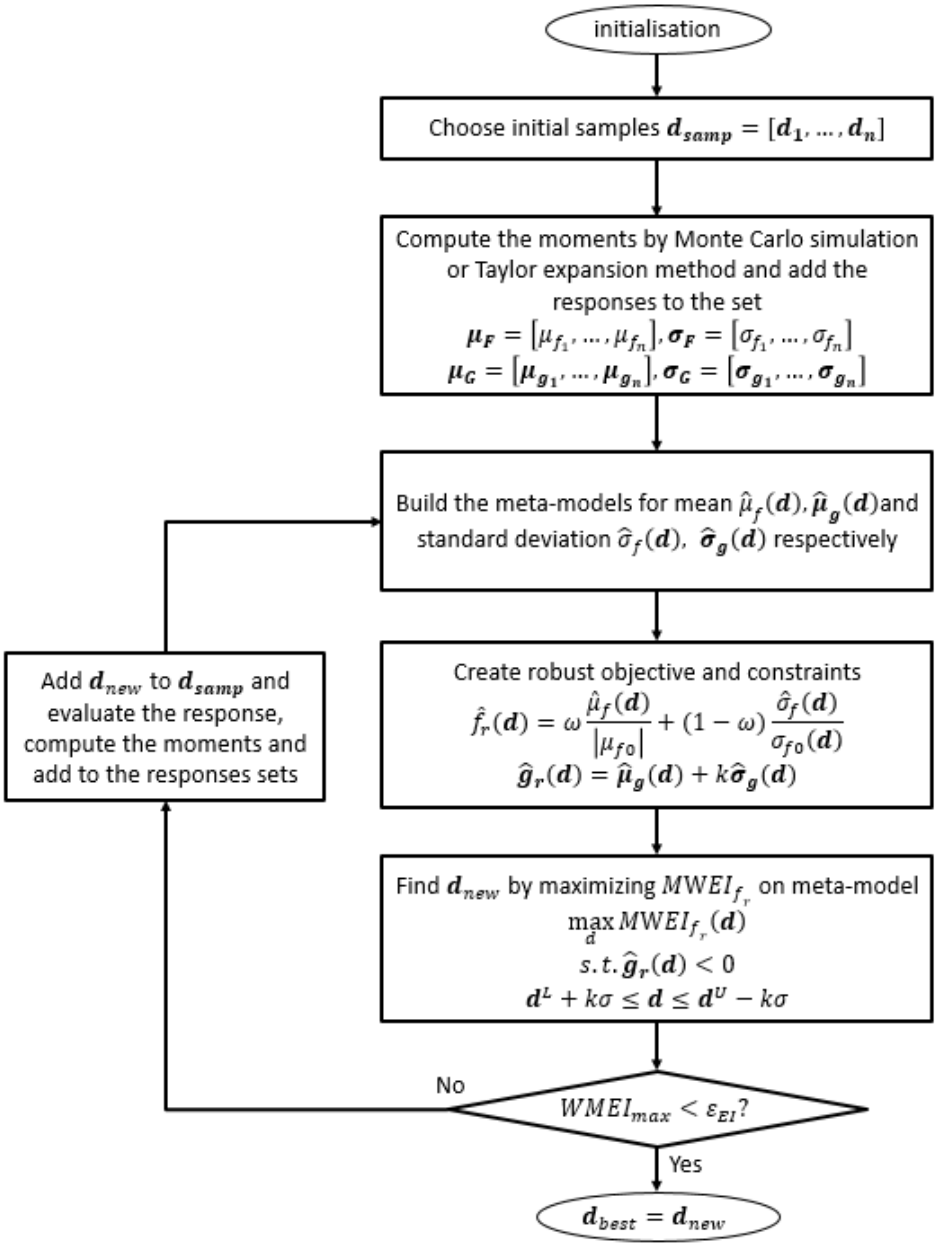


Fig. 2.12. The flowchart of proposed RDO algorithm with meta-model

However, this strategy needs a great number of evaluations in order to compute the moments. During the uncertainty propagation, neither the Taylor expansion nor Monte Carlo simulation can get sufficient information within one evaluation. The former method needs to calculate the gradient and/or hessian information, the latter one needs frequently 10^6 samples which seems impossible for the heavy models.

For the purpose of reducing the number of evaluations especial when the model is too heavy, we decide to create another meta-model for the original objective function and constraints. Thus, after the initial samples have been chosen and evaluated, meta-models $\hat{f}$ and $\hat{\boldsymbol{g}}$ are built. With this first layer of meta-model, uncertainty propagation methods can be used to calculate the requisite moments without taken too much time. Also using first order Taylor based method for example, the calculation of moments change to:

$$\begin{cases} \hat{\mu}_f = \hat{f}(\boldsymbol{d}) \\ \hat{\sigma}_f^2 = \sum_{i=1}^{p} \left(\frac{\partial \hat{f}(\boldsymbol{d})}{\partial d_i}\right)^2 \sigma_i^2 \end{cases} \tag{2.66}$$

$$\begin{cases} \hat{\mu}_{g_j} = \hat{g}_j(\boldsymbol{d}) \\ \hat{\sigma}_{g_j}^2 = \sum_{i=1}^{p} \left(\frac{\partial \hat{g}_j(\boldsymbol{d})}{\partial d_i}\right)^2 \sigma_i^2 \end{cases} \tag{2.67}$$

Then the second layer of meta-models: the means $\hat{\mu}_f, \hat{\boldsymbol{\mu}}_{\boldsymbol{g}}$ and standard deviations $\hat{\sigma}_f, \hat{\boldsymbol{\sigma}}_{\boldsymbol{g}}$ are built. Once the robust formulation is created by these surrogate moments, the left part is the same as the former method. This modified method is called double-layer method as it requires two kriging models in different usages: one for calculate the moments and the other to create the robust formulation. As a contrast, the former method is called single-layer method. So, the flowchart of this double-layer method changes to:

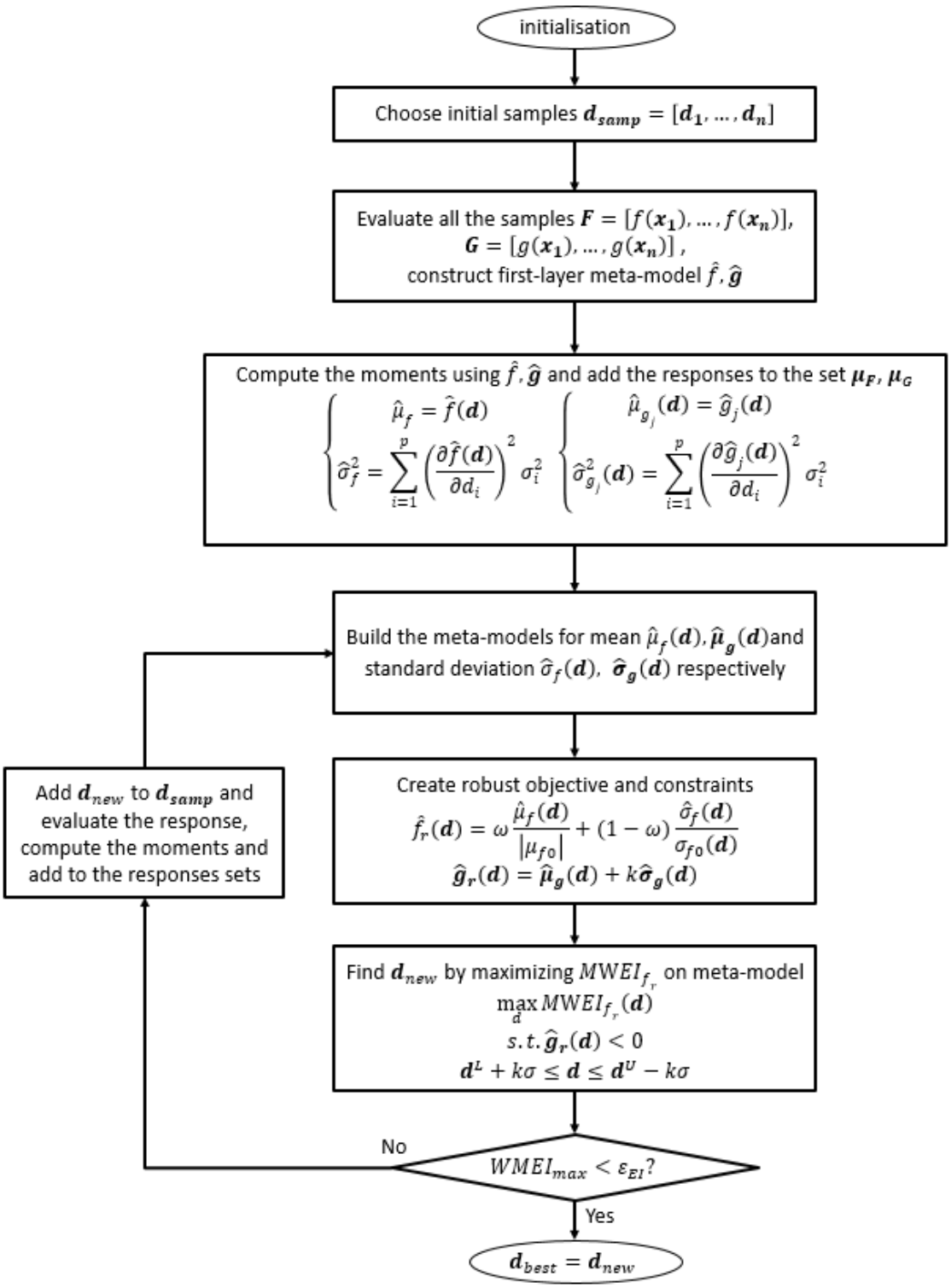


Fig. 2.13. The flowchart of proposed double-layer kriging RDO algorithm

2.2.2 Mathematical example

The two above algorithms are compared on the mathematical model as shown below in the figures. Formulation 7 in Table 1.3 in Chapter 1 is used here as an example:

$$f_r(\boldsymbol{d}) = \omega \frac{\mu_f(\boldsymbol{d})}{|\mu_{f_0}|} + (1-\omega)\frac{\sigma_f(\boldsymbol{d})}{\sigma_{f_0}} \tag{2.68}$$

The weight ω of the robust formulation is chosen as 0.5 at first and the means and standard deviations are calculated by the first order Taylor expansion. μ_{f_0} and σ_{f_0} can be obtained by any initial sample, here for comparison, we fix them with the values calculated by the point [3.5; 5].

Fig. 2.14 is the meta-models and samples with single-layer kriging, and Fig. 2.15 is with double-layer kriging. Both approaches starts with 20 random initial samples. In the figures, colored contours are for initial objective function, blue dotted lines are for the initial constraints and black solid lines denote meta-models of robust constraints. Green points are the 20 initial samples to build the first meta-models, red points are added in each iterations. The cyan lines in the figure 2.15 are the limit-state of initial constraints computed by the meta-models.

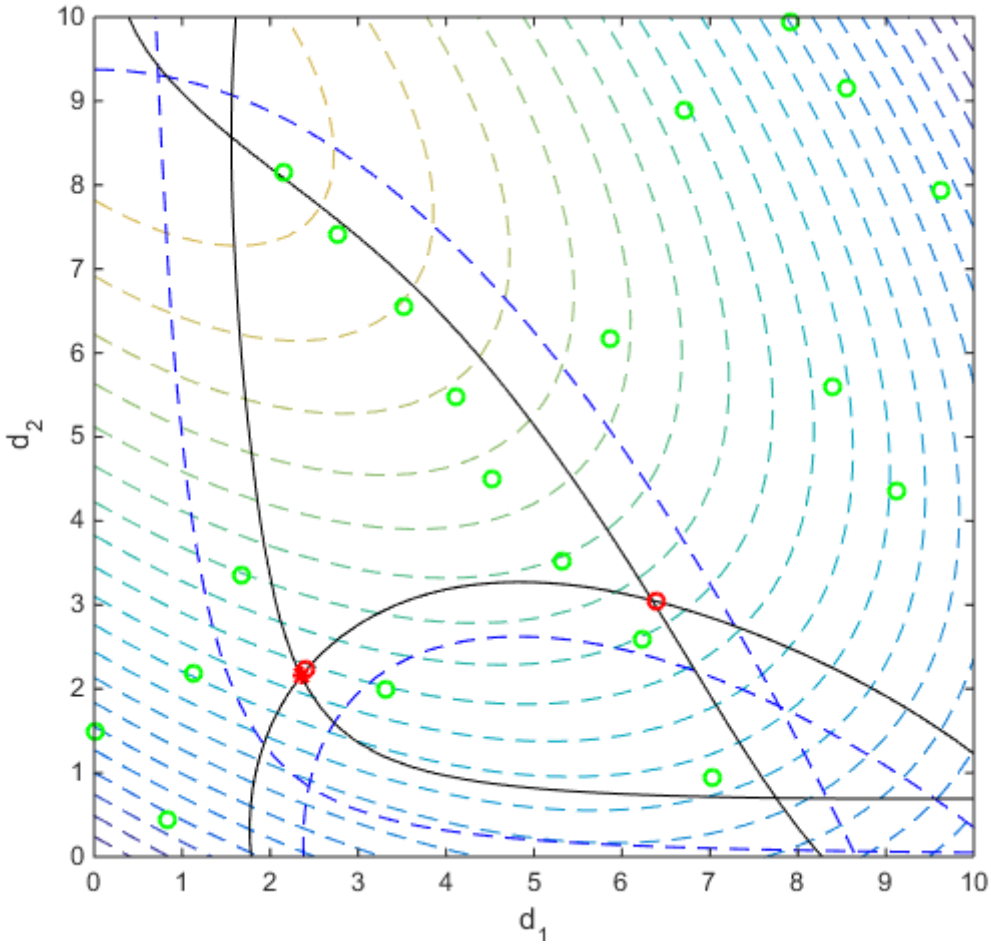


Fig. 2.14. Final iteration of adaptive single-layer kriging based RDO with $\omega = 0.5$

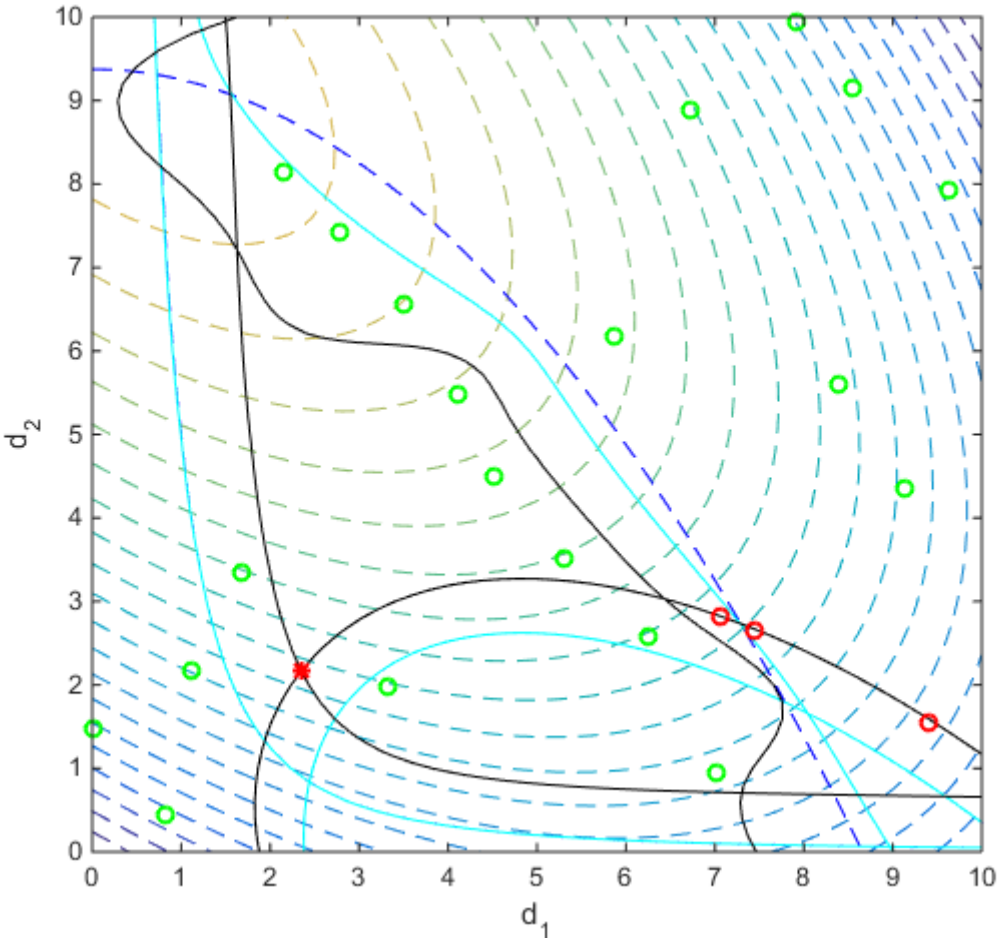


Fig. 2.15. Final iteration of adaptive double-layer kriging based RDO with $\omega = 0.5$

Table 2.4. Result of adaptive kriging based RDO with different value of weight

Method	Number of iterations	Number of evaluations	Optimum found with meta-model	Optimum found with mathematic model
Single-layer kriging	4	$20 + 12 \times 4$	[2.3523; 2.1565]	[2.3524; 2.1568]
Double-layer kriging	6	$20 + 6$	[2.3523; 2.1562]	

From the above table it can be seen that double-layer kriging method can reduce the number of evaluations compared to single-layer kriging method, but the optimum found is further from the optimum found with mathematic model. It is reasonable since two layers of meta-model bring one more approximation in the system and increase the error. However, with this example, the difference of optima found by these two methods is small and the lost accuracy is acceptable. Moreover, in this example, only first order Taylor expansion is used, if the second derivative is calculated also or the Monte Carlo simulation is introduced to calculate the moments, it is foreseeable that the optima will be more precise and also the gap of evaluations numbers between these two approaches will be further widened.

In order to check the ability of finding the robust solution with different requirements, $\omega = 0$ and $\omega = 1$ are also tested with dual-kriging method. The meta-models and samples are shown in the following figures and the table presents the numerical results.

Table 2.5. Result of double-layer kriging based RDO with different value of weight

Value of weight	Number of evaluations	Optimum found with meta-model	Optimum found with mathematic model
1	5	[2.3497; 2.1673]	[2.3524; 2.1568]
0.5	6	[2.3523; 2.1562]	[2.3524; 2.1568]
0	2	[1.5842; 8.2851]	[1.5828; 8.3685]

As can be seen from the figures and table, this approach can get a quite accurate solution with a small number of evaluations. Besides, it can adjust the focus of searching by the weight parameter. If the requirement prefers to have a smaller mean value, then the weight is close to 1. On the contrary, if the requirement prefers to have a smaller standard deviation value, the weight is close to 0.

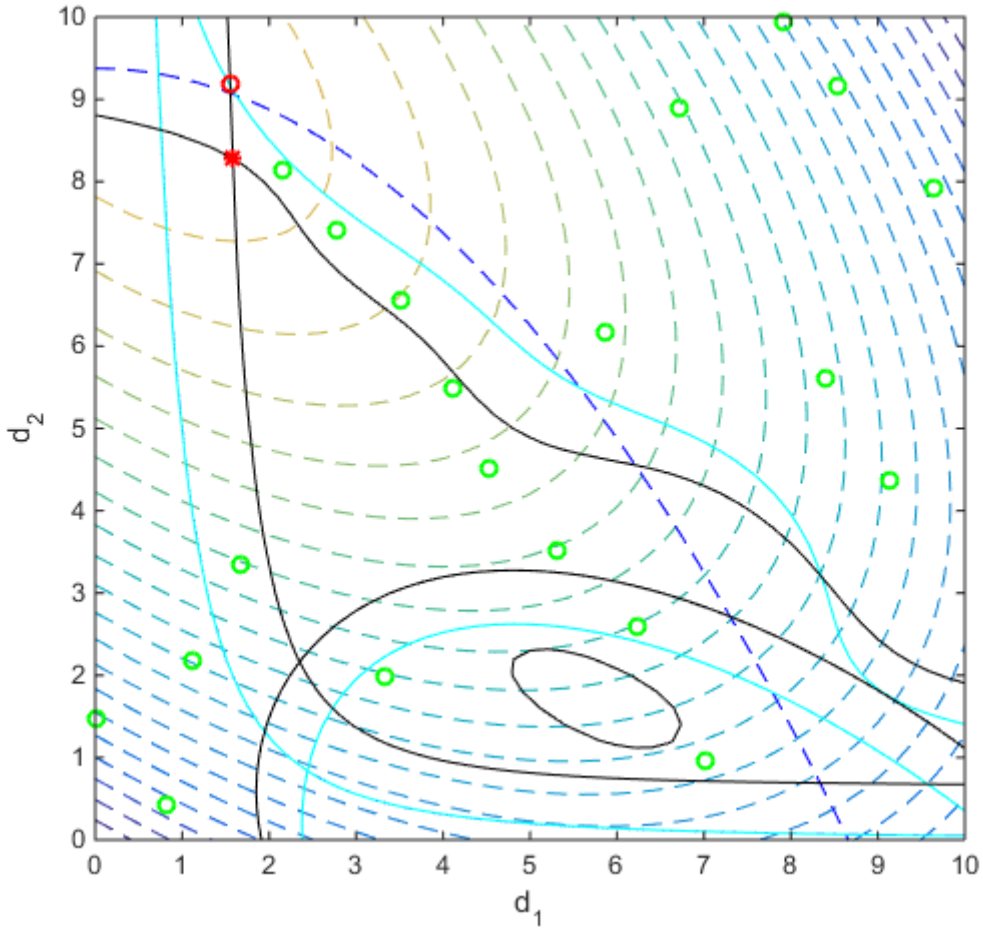


Fig. 2.16. Final iteration of adaptive double-layer kriging based RDO with $\omega = 0$

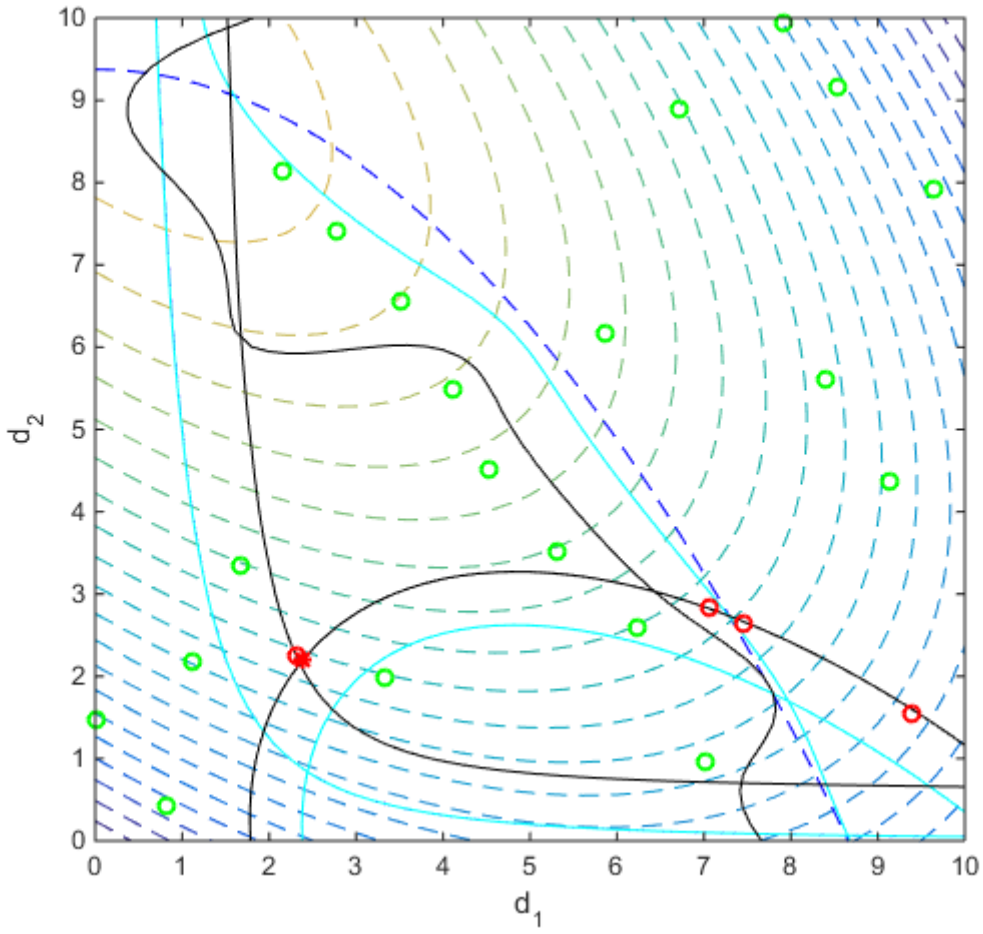


Fig. 2.17. Final iteration of adaptive double-layer kriging based RDO with $\omega = 1$

2.3 Adaptive Kriging based RBDO

For Kriging based RBDO, the strategy is very different from RDO. Attaining a highly accurate meta-model for the global design space is however not necessary, nor desired for the RBDO methods. Instead of wasting computational effort for sampling the non-optimal zones of the design space, the focus should be on promising areas with a high chance of detaining the optima. Therefore, the idea of the strategy is to simultaneously seek the optimal solution (exploitation) while improving the global prediction accuracy of the meta-model (exploration).

In addition, with the aim of searching the global deterministic optimum, locating the global optimum is more important than improving the accuracy of the kriging model. However as EI is highly multimodal, a local minimum is often found. To be sure to find the global solution, more attention should be paid on the infill criterion. Thus, the MWEI criterion present in section 2.2.1 in this Chapter could be used here also as we can adjust the searching process to pay more attention to finding global deterministic optimum by changing the value of weight. As the advantages of MWEI has been explained before, it

will not be detailed here and for each type of RBDO method, new strategies will be presented in following parts.

2.3.1 Infill Strategies for Double-loop method

Double-loop method like Performance Measure Approach (PMA) (Tu, Choi, & Park, 1999) has a nested structure: The outer loop seeks for the optimum and the inner loop analyzes the reliability by searching the Most Performance Target Points (MPTPs) by solving an optimization problem.

There are two places where ISC can be introduced to improve the accuracy of the kriging model: Outer loop and inner loop. As outer loop is an optimization with inequality constraints, the criterion MWEI and the meta-models of constraints can be used.

$$\begin{aligned} \boldsymbol{d}^k &= \underset{\boldsymbol{d}}{\operatorname{argmax}}\, MWEI_f(\boldsymbol{d}) \\ s.t.\ \ &\hat{g}(\boldsymbol{x}^k) \leq 0 \end{aligned} \tag{2.69}$$

where $\boldsymbol{d}^k$ is the optimum found in kth iteration, $\hat{g}(\boldsymbol{x}^k)$ is the predictor of the maximal performance measurement for constraint calculated in the inner loop.

For the inner loop, EI is used directly with the original constraint because this last is an explicit function of design variables.

$$\begin{aligned} \boldsymbol{x}^k &= \underset{\boldsymbol{x}}{\operatorname{argmax}}\, EI_g(\boldsymbol{x}) \\ s.t.\ \ &\|(\boldsymbol{x} - \boldsymbol{d})/\sigma\| = \beta_t \end{aligned} \tag{2.70}$$

where β_t is the target index of reliability depending on the chosen probability of failure, and σ is the standard deviation of the random variables X. Meanwhile as the inner loop intents to maximize the performance subject to the target reliability constraint, the chosen best value is the maximum among all samples on the hypersphere and the sign before g is opposite than the one before f.

$$EI_g(\boldsymbol{x}) = \begin{cases} [\hat{g}(\boldsymbol{x}) - g_{max}]\Phi(z_g) + \hat{s}_g(\boldsymbol{x})\phi(z_g) & \text{if } \hat{s}_g > 0 \\ 0 & \text{if } \hat{s}_g = 0 \end{cases} \tag{2.71}$$

where $z_g = [\hat{g}(x) - g_{max}]/\hat{s}_g(\boldsymbol{x})$, g_{max} is the maximum sampled constraint value on the hypersphere at the current iteration. Whereas a satisfied g_{max} may not exist at the beginning, it is initialized with $g(\boldsymbol{d})$ to be sure that there exist points on the hypersphere with greater values. After one iteration, a solution of Equation 2.53 is found and g_{max} is set to $\hat{g}(\boldsymbol{x}^*)$.

However, as the two loops are nested, the enrichment in inner loop may bring out thousands of model evaluations. To test it, two strategies are proposed: the first one (PMA1) adds new samples only inside the outer loop, whereas the second (PMA2) enriches inside both the outer and inner loops.

2.3.2 Infill Strategies for Single-loop method

For Single loop method like Single Loop Approach (SLA) (Liang, Mourelatos, & Tu, 2004), the main point is that the model in the inner loop is replaced by an approximation based on a first order Taylor expansion. The probability of failure is then approximated to avoid the numerous evaluations required for reliability analysis. Thus, MWEI is used straightly in this optimization with this time x^k is calculated by an approximation:

$$\boldsymbol{x}^k = d + \beta_t \frac{\sigma \circ \nabla g(\boldsymbol{d})}{\|\sigma \circ \nabla g(\boldsymbol{d})\|} \circ \sigma \tag{2.72}$$

It is important to note that due to its first order approximation, the method itself has already loose some precision. Therefore, it is expected that with a surrogate model, the two approximations will be superposed and the accuracy will be further reduced.

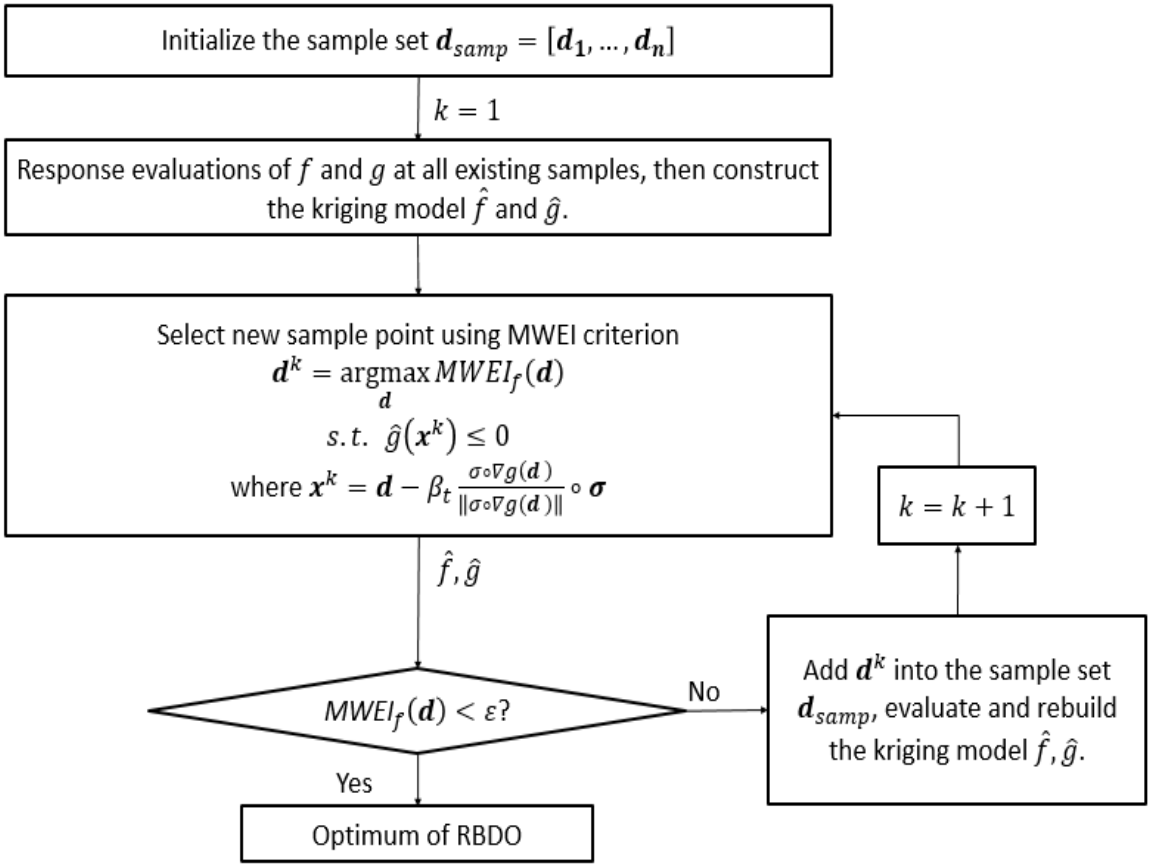


Fig. 2.18. The process of infill strategy with SLA

2.3.3 Infill Strategies for Sequential method

Sequential decoupled method like Sequential Optimization and Reliability Assessment (SORA) (Du & Chen, 2004) are based on a series of sequential deterministic optimizations and reliability assessments. The main point is to shift the boundaries of constraints to the feasible direction based on the reliability information obtained in the former iteration. The first deterministic optimization aims at searching the global optimum. Reliability

assessments are conducted after to locate the MPTP which corresponds to the desired probability of failure. Then new optimization is solved by taking into account the shift computed with MPTP.

Three strategies are proposed. As the key point of deterministic optimization process is to find the global optimum, MWEI and meta-model of constraints are used for the first strategies (SORA1) to add new points during each deterministic optimization.

$$\begin{aligned} \boldsymbol{d}^{k+1} &= \underset{\boldsymbol{d}}{\operatorname{argmax}}\, MWEI_f(\boldsymbol{d}) \\ s.t.\ \ &\hat{g}(\boldsymbol{d} - \boldsymbol{t}^{k+1}) \leq 0 \end{aligned} \tag{2.73}$$

where k is the iteration number, $\boldsymbol{t}^{k+1}$ denote the shift from the former $\boldsymbol{d}^k$ to the MPTP $\boldsymbol{x}^k$ calculated by the reliability assessment for the constraint in the k-th iteration:

$$\boldsymbol{t}^{k+1} = \boldsymbol{d}^k - \boldsymbol{x}^k \tag{2.74}$$

For the first iteration, the initial value of $\boldsymbol{t}$ is 0.

Then for reliability assessments, EI and the surrogate constraints for ISC sub-problems as Equation (2.46) are applied. The solutions on a hypersphere around $\boldsymbol{d}^k$ are added to enrich points in order to find a more precise MPTP. This enrichment improves the accuracy of local constraint boundaries in the vicinity of the current design point. It is more rational and less expensive than improving the accuracy for the whole domain.

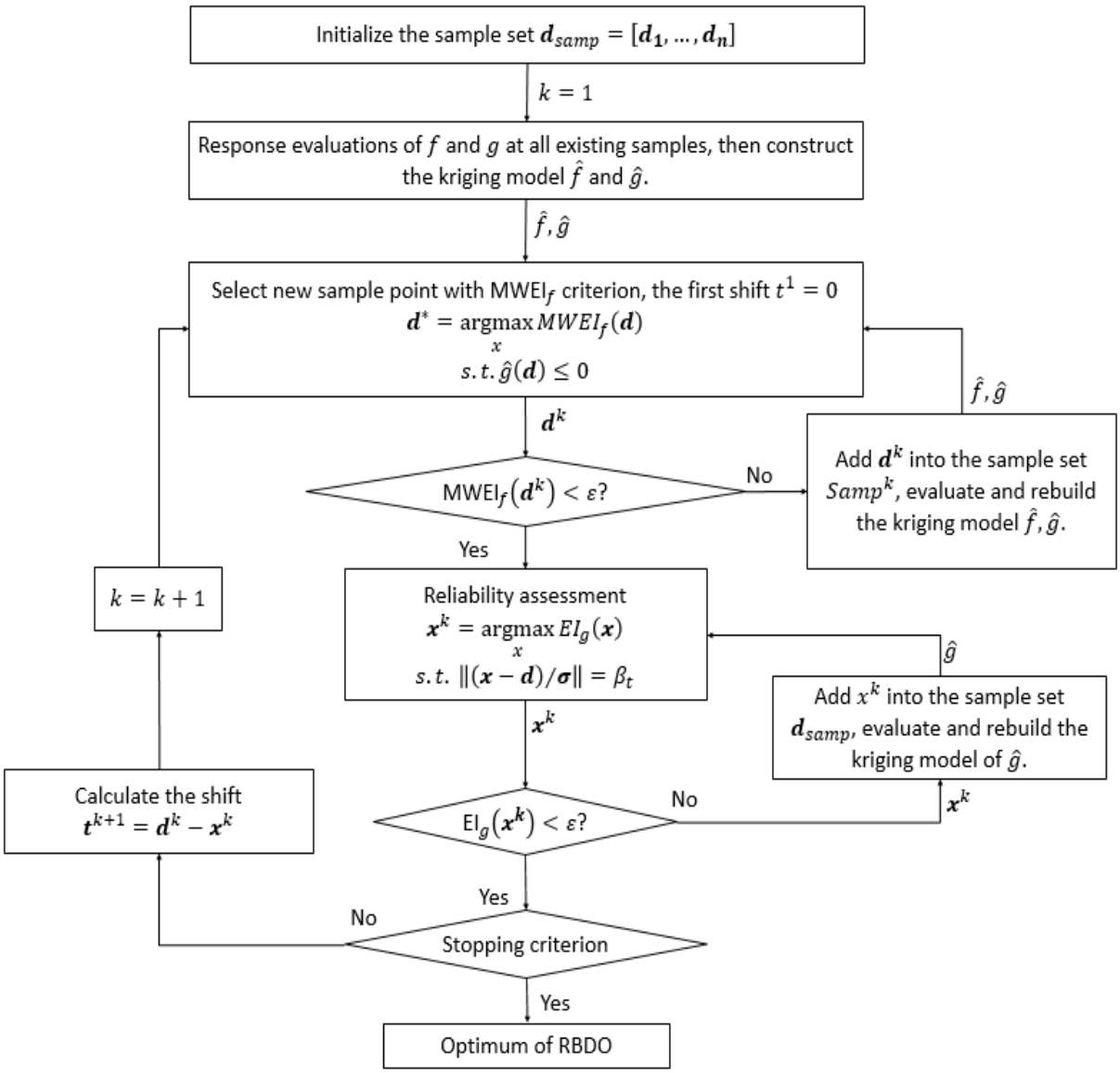


Fig. 2.19. The process of SORA1 strategy

The second strategy (SORA2) differs from the first one by the fact that enrichment of the kriging models with MWEI criterion is made at first iteration only. For all other iterations, the deterministic optimization is made with the meta-models $\hat{f}$ and $\hat{g}$.

Indeed, it seems to be more important to add samples on the constraints boundaries and the solution of the optimization at all iterations except first one for which the solution is distant from those boundaries. So only the first iteration needs to use the ISC to insure that the global optimum can be found accurately by adding some new samples.

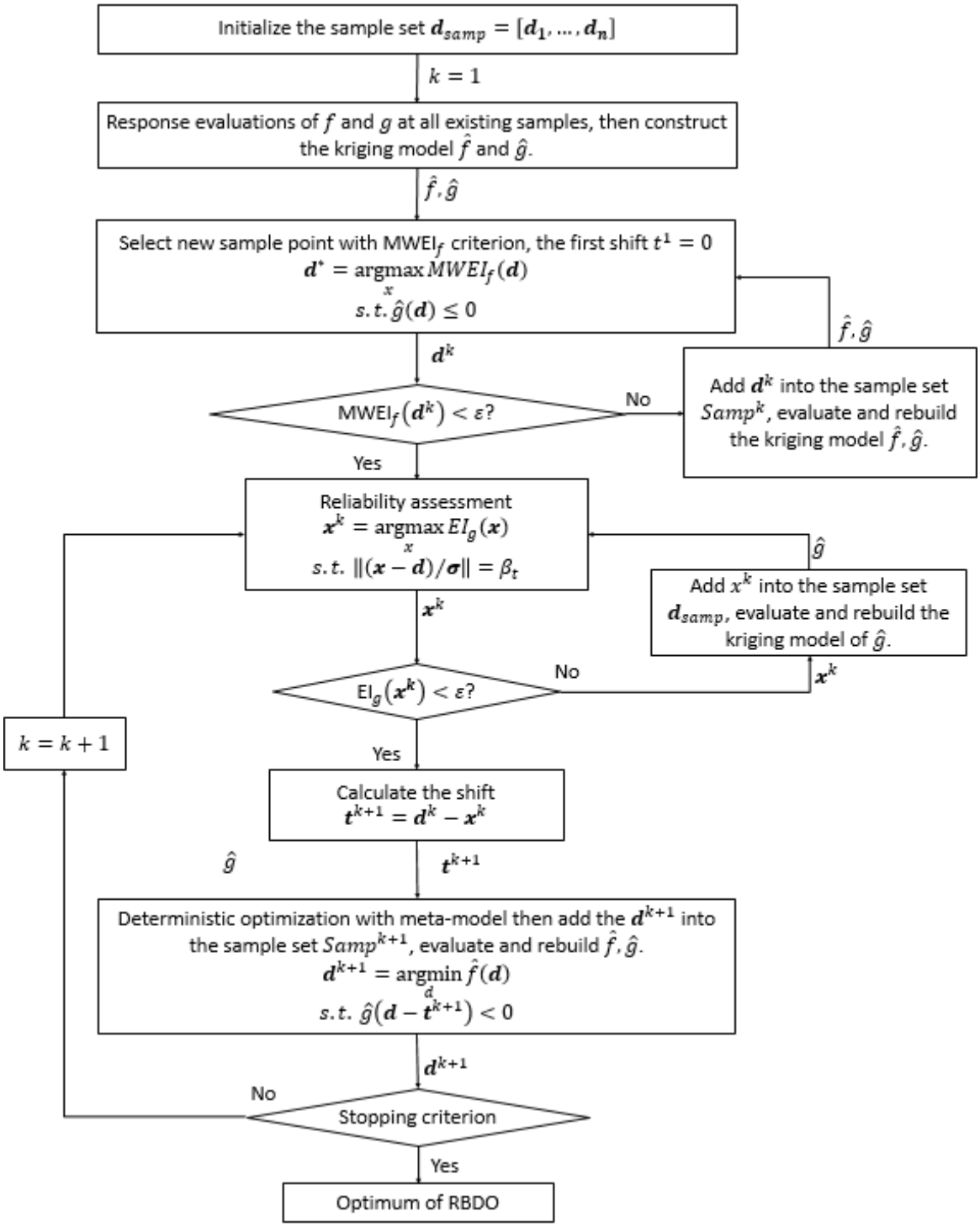


Fig. 2.20. The process of SORA2 strategy

For the third strategy (SORA3), if the deterministic optimum found in k-th cycle is close to any of the other k-1 cycles, as the former reliability assessments have already added points in this region, the accuracy is considered to meet the requirement so there is no need to add new samples. A proximity criterion defined as in next expression can be introduced before each reliability assessment:

$$\left\|\left(\boldsymbol{d}^k - \boldsymbol{d}^i\right)/\sigma\right\| < \beta_t, \quad i = 1, \dots, k-1 \tag{2.75}$$

where $\boldsymbol{d}^i$ is the deterministic optimum found by the ith cycle. If the above criterion is satisfied, the meta-model will be used directly and only MPTP are evaluated. For the parts of deterministic optimization, it takes the same strategy as the second one. The flowchart of SORA3 is shown in Fig. 2.21.

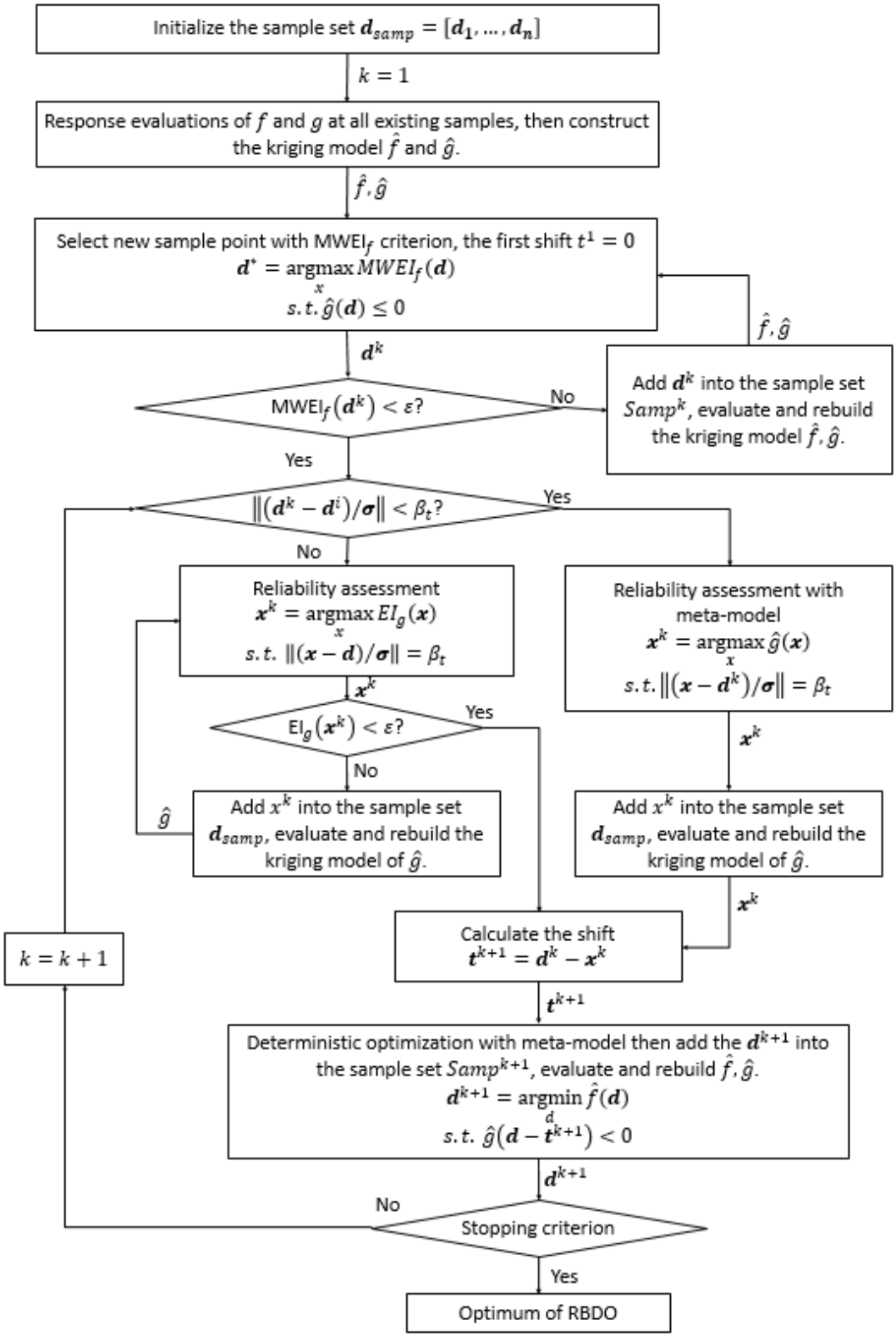


Fig. 2.21. The process of SORA3 strategy

2.3.4 Mathematical Example

To assess the efficiency of adaptive kriging-based RBDO methods, also the analytical problem in (Dong-Wook Kim, Nak-Sun Choi, Choi, & Dong-Hun Kim, 2015) with two variables and three constraints is analyzed. Noting that the random variables comply with normal probabilistic distributions and the standard deviations are all equal to 0.3. Lower and upper bounds are 0 and 10 respectively for both variables. The target reliability index β_t is chosen as 2, so that the target probability of failure $P_t = \Phi(-\beta_t) = 2.28\%$.

The results are given in Table 2.3 with an initial sampling of 20 points. The probability of failure P_f is calculated by Monte-Carlo simulation with 10^6 samples. For comparison purpose, results given by RBDO methods with the original problem are also presented. All the iterative kriging-based RBDO methods lead to a reduced number of evaluations. SLA with kriging is not accurate enough, the maximum probability of failure is much greater than P_t because of the approximation used to simplify the reliability analysis. As expected, PMA with infill during inner loop requires thousands of samples to evaluate. The other PMA strategy is faster but as it does not add samples in the vicinities of MPTP, the accuracy is not sufficient. Kriging-based SORA strategies lead to the best results and the third one is the most efficient. Fig. 2.22 shows the different iterations of SORA3, the contours are for objective function, dashed lines and solid lines present real objective or constraints and surrogate ones, respectively.

Table 2.6. Result of mathematical example using different strategies or RBDO

Strategy	Number of evaluations	Optimal solution	Optimal value	Maximal P_f (%)
SLA (exact model)	165	[2.2512; 1.9677]	-1.9953	2.32
PMA/SORA (exact model)	3183/531	[2.2513; 1.9691]	-1.9945	2.27
SLA	26	[2.2466; 1.9617]	-1.9996	2.59
PMA1	29	[2.2494; 1.9649]	-1.9972	2.44
PMA2	1804	[2.2513; 1.9691]	-1.9945	2.27
SORA1/2/3	142/97/45	[2.2513; 1.9691]	-1.9945	2.27

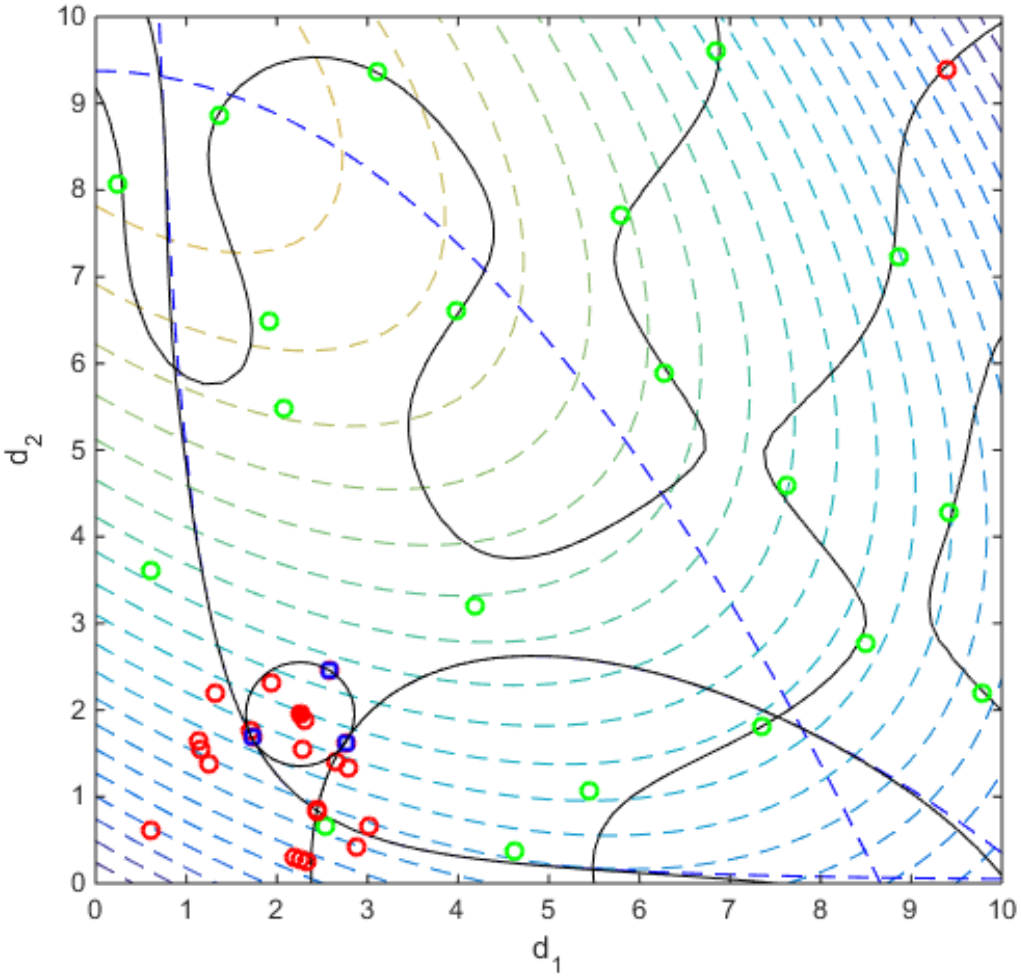


Fig. 2.22. The iterations of SORA3 for the mathematical example

2.4 Adaptive Kriging based RBRDO

This study is quite recent almost all the researches are in the last five to ten years. (Lai et al., 2016) proposed a sequential approximate strategy with Kriging meta-model for RBRDO. However the authors did not use any ISC, instead new points are selected as the solution of optimization with surrogate objective function and constraints only. Thus, the method does not guarantee that the added points can improve the accuracy of the meta-model as much as possible or are the optimal solution of the meta-model.

(Arsenyev, Duddeck, & Fischersworring-Bunk, 2015) decided to build two levels of meta-models. After evaluating the generated samples, the first level Kriging meta-model are fitted for the initial objective function and constraints. Then authors choose double-loop methods to analyze the reliability and assess the robust objective function using Monte-Carlo sampling for each training point. Then, the second layer Kriging meta-model is fitted over the design space using robust objective values and constraint values at MPTP from PMA, which are now the deterministic constraints for the outer-loop optimizer. These surrogates share the points with the first layer surrogates which enforces simultaneous refinement of the approximations in the design space for both layers. The

EGO step with EI and EF as ISC is performed on the second level surrogate, searching for the promising points to minimize robust objective while satisfying reliability constraints. For each infill points found by EGO step, the values of the uncertain variables are determined to refine the surrogates in the uncertain subspace. In case of robust objective function, the point that maximizes the mean square error is selected. For the constraints, the PMA analysis is performed and if the predicted probability of constraint being active is higher than a threshold value, the obtained point is added to the constraint's training set. When all infill points for all GP surrogates are found and evaluated, the surrogates are refitted and the next iteration of the method is performed. If highest EI of infill points for last two iterations is less than a threshold value, the iterations are stopped. The flowchart of the proposed methodology is:

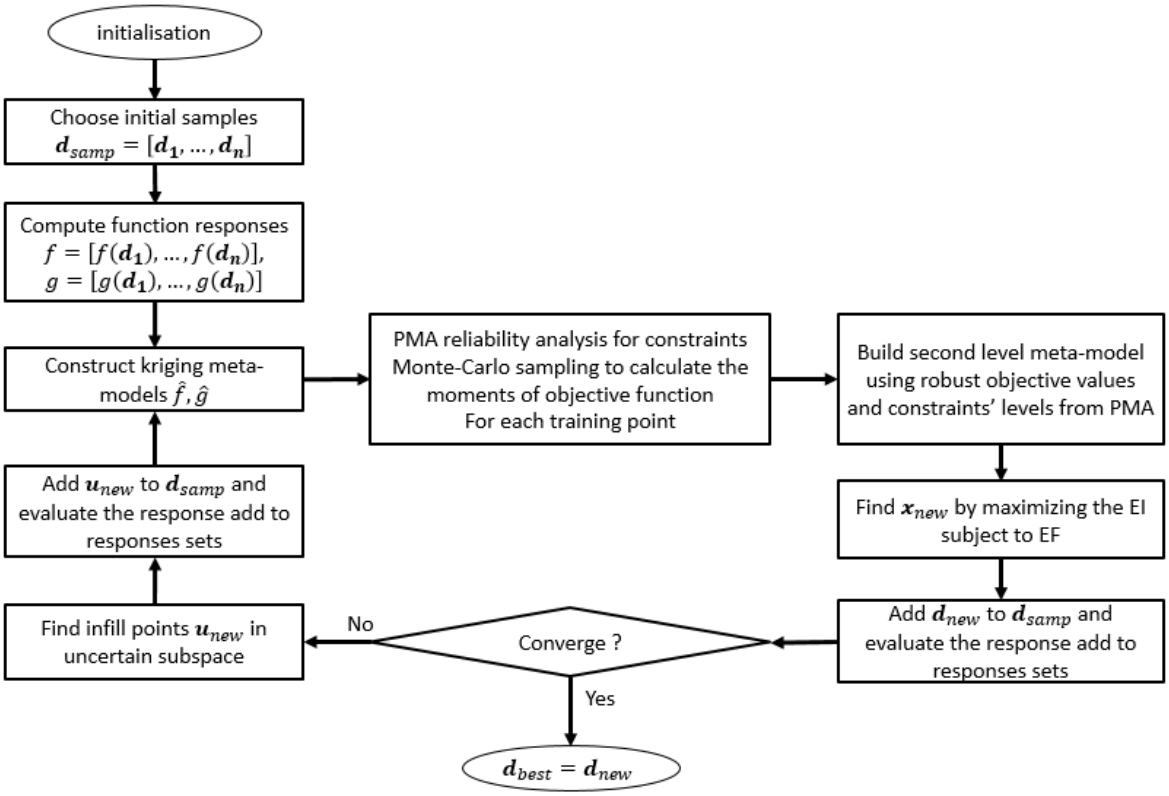


Fig. 2.23. Flowchart of the methodology proposed by (Arsenyev et al., 2015)

However, in this approach, to build the second layer meta-model, each sample point will be treated with PMA and Monte-Carlo sampling method, thus augmenting the computational burden greatly.

To simplify the methodology, new strategies are proposed in this chapter and as there are three types of method to analyze the reliability (double-loop, single-loop and decoupled sequential), each type of method with robust formulation of objective function will be tested to pick out the most efficient one.

2.4.1 New infill strategy for RBRDO

As traditional RBRDO is the combination of RDO and RBDO, adaptive Kriging based RBRDO can also be combined with adaptive Kriging based RDO and RBDO. So that we use the strategies presented in the RBDO section and append the formulation introduced in previous chapter.

First, for the single-loop method, adaptive kriging based SLA method presented in section 2.3.2 can be used, then for considering the robust objective function, double-layer kriging method presented in section 2.2.1 is also needed. One meta-model is for initial objective function, and the other surrogate the robust objective function with moments calculated by the first one. The flowchart is as follow:

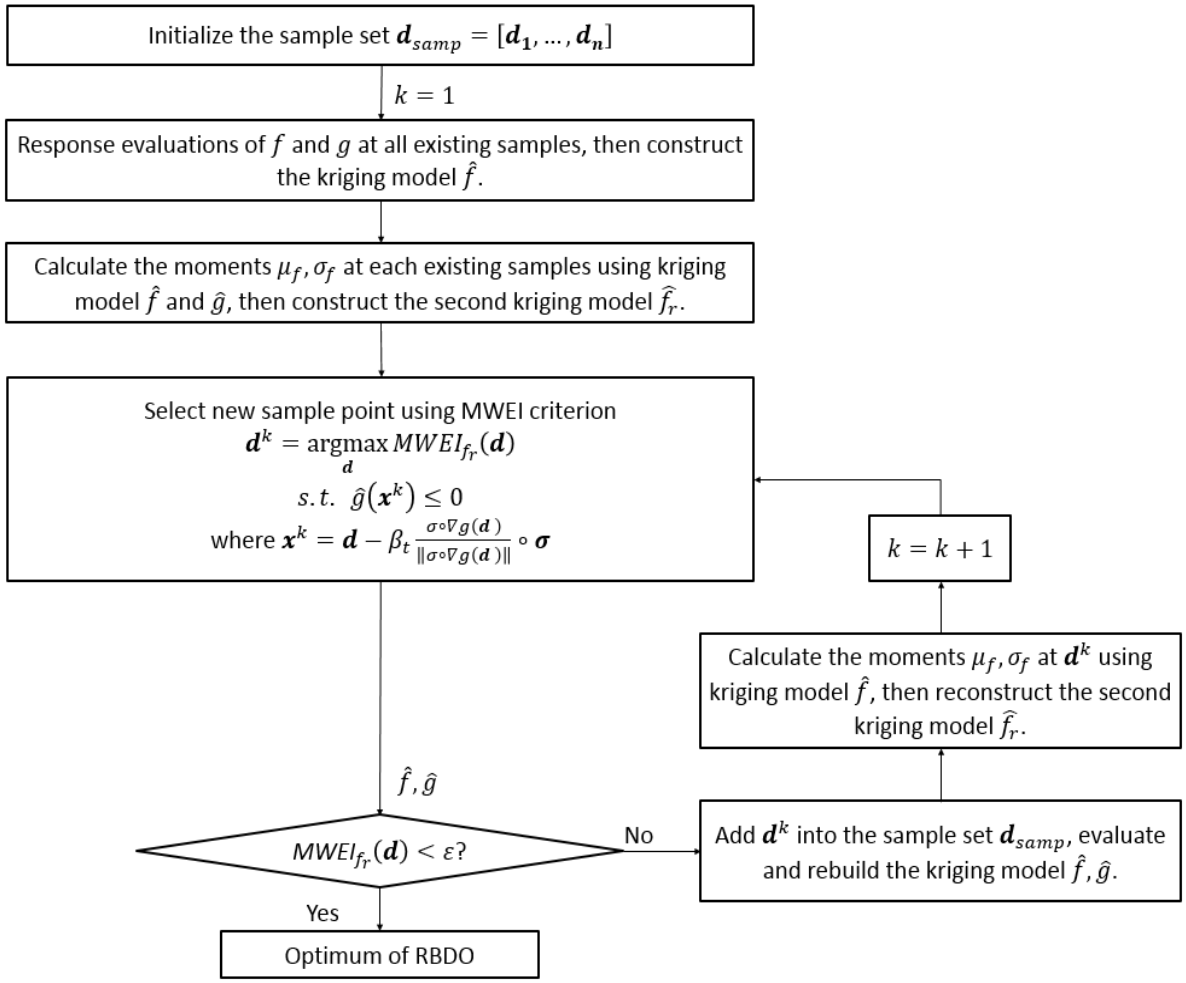


Fig. 2.24. Flowchart of the double-layer kriging based RBRDO using single-loop method

Also, for the double-loop method, as has been discussed in section 2.3, the strategy PMA1 requires much less evaluations, as our main purpose is to decrease the computational burden for heavy models, so that PMA1 is chosen here to combine with double-layer kriging based RDO method. Thus, the enrichment only occurs in outer loop.

At last, for sequential decoupled method, SORA3 is the most efficient approach and is used with Normal-the-Best formulation here as an example to compare all the three mentioned methods. As in Chapter 1, the target f_t is chosen as the minimum of DDO - 2.6266 as an ideal result which cannot be attained, and μ_{f_0}, σ_{f_0} are fixed as the values computed by the design point [3.5; 5].

2.4.2 Mathematical example

Results for the mathematical example is shown in the following figures and table. Colored contours are for deterministic objective function, blue dotted lines are for deterministic constraints and black lines are meta-model of constraints. Green points present 20 initial samples, red points are added during iterations. The blue points in SORA present the maximum performance target points.

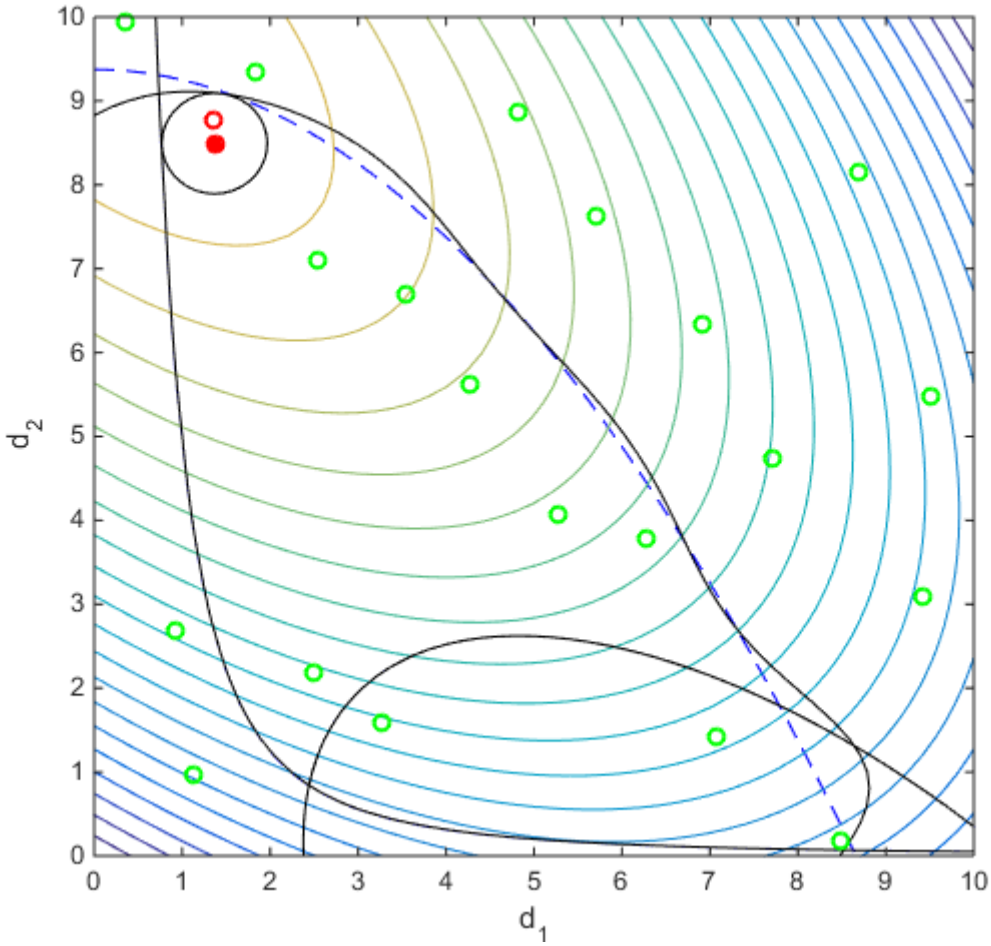


Fig. 2.25. Final iteration of adaptive double-layer kriging based RBRDO with SLA

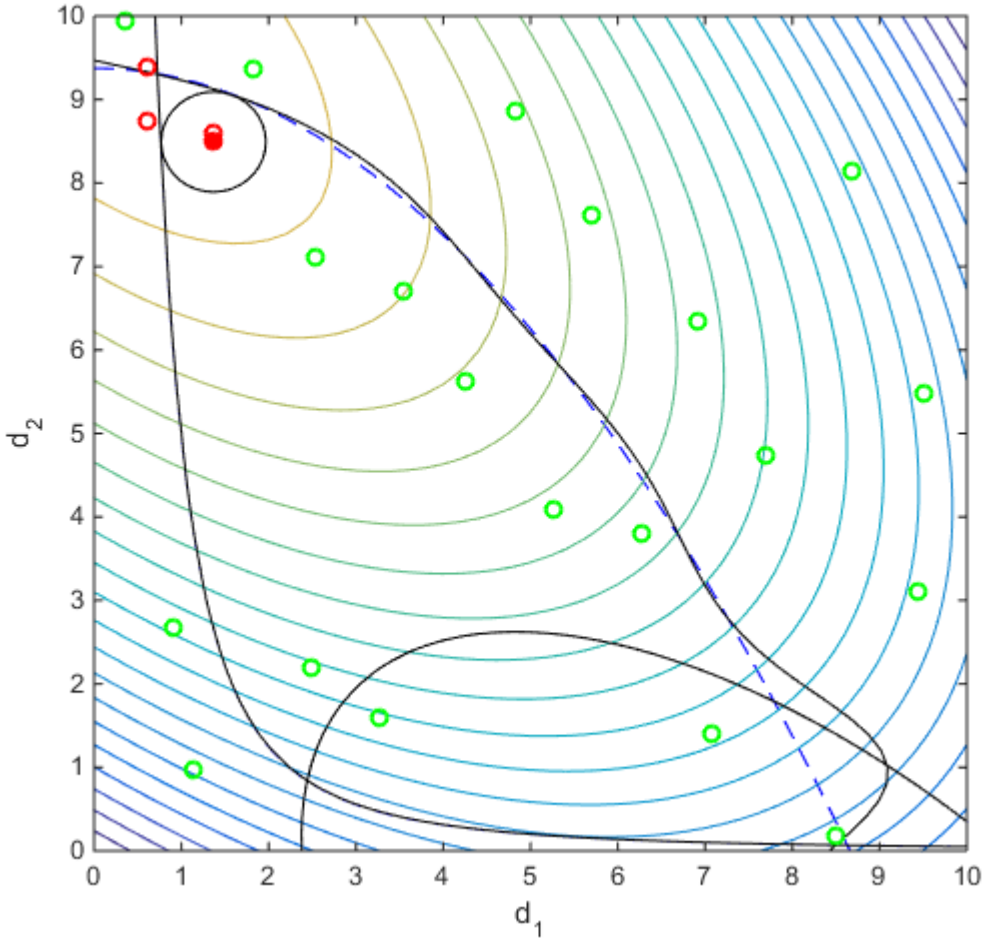


Fig. 2.26. Final iteration of adaptive double-layer kriging based RBRDO with PMA1

From the results, it can be seen that SORA3 with double-layer kriging method takes the most time and SLA double-layer kriging strategy is not accurate enough. SORA3 gets the solution the closest to the one found by using the mathematical model without building a meta-model. However, for RBRDO the accuracy is reduced because of the double-layer kriging. The robust objective meta-model is built based on moments calculated by the other surrogate model, so that it lose some precision.

Table 2.7. Result of mathematical example using different strategies of RBRDO

Strategy	Number of iterations	Number of evaluations	Optimal solution
Double-layer kriging + SLA	3	20+3	[1.3539; 8.7980]
Double-layer + PMA1	5	20+5	[1.3670; 8.5068]
Double-layer + SORA3	5	20+31	[1.3672; 8.5042]
Mathematical model	--	612	[1.3675; 8.5018]

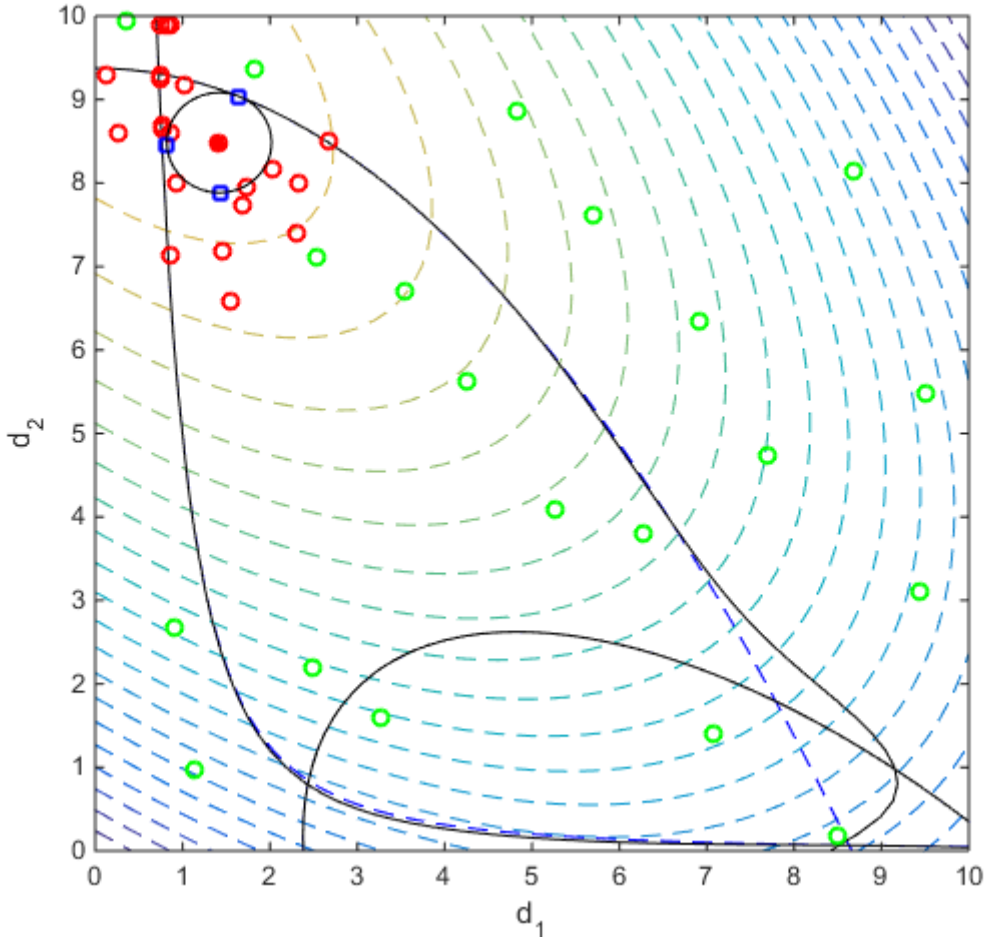


Fig. 2.27. Final iteration of adaptive double-layer kriging based RBRDO with SORA3

3 Conclusion

This chapter mainly describes approaches of adaptive kriging based optimization with uncertainty for heavy models. This research topic is gaining importance during the last decades. Many researchers proposed their methods to reduce the computational burden but each method has its own limitation.

We try to propose strategies that focus on where to add new samples and with which ISC for every different type of optimization with uncertainty. Although the suitable strategies for each type is disparate, the purposes are always trying to use meta-model to avoid time consuming evaluations and using adapted ISC to add new samples at the most needed positions.

With a simple mathematical example, different strategies are tested in order to choose the most effective ones.

For WCO, meta-models are built on the original objective function and constraints, then two criteria are proposed to estimate the worst-case values. After comparing their

solutions and numbers of evaluations, it can be seen that the one enrich with WCEI is more accurate.

Then for RDO, as the calculations of moments of original objective and constraints need large amount of evaluations especially when simulation methods like Monte Carlo approach is used. So as to reduce the time consumption, double-layer kriging meta-model is proposed. The first layer is building surrogate models for original objective and constraints, then second layer meta-models of moments are built with the former layer, formulations and ISC like MWEI can be used with this estimated moments to solve the problem. Results of the mathematic example show that this strategy is faster than single-layer meta-model method without losing much accuracy.

Different approaches using SLA, PMA and SORA separately are proposed for RBDO. Among them, the strategy which uses MWEI for first deterministic optimization and a proximity criterion before reliability assessments is the most efficient one.

For RBRDO, double-layer kriging method combines with SORA presents a good compromise between accuracy and speed of convergence.

However, these methods are proposed for heavy models, a simple mathematical model cannot make too much sense. Thus in the next chapter, a more complex electromagnetic device that is a safety isolating transformer will be introduced and its analytical model and finite element model will both be used to compare these approaches.

Chapter 3: Transformer

The safety transformer is one of the most used electrical devices thanks to the multiplicity of its various applications and its simple manufacture (Tran, 2009). This device exhibits other interesting features such as:

- Multiphysics
- Implicit equations
- Highly constrained
- Badly scaled design variables, objectives and constraints
- Multimodal

So as a typical electromagnetic device with all above difficulties, it is chosen in this manuscript to compare the efficiency of the former proposed methods.

The device is a single-phase safety isolating transformer at 50/60Hz whose operation is reversible. The power ranges from a few dozens of volt-amperes to thousands of volt-amperes. The primary side voltage can be 230V, 400V or maximum 690V, and the secondary side voltage can be 24V or 12V. This is a low power transformer designed for installation in electric cabinet.

It is manufactured with grain-oriented E-I laminations. In order to reduce the eddy current losses, the magnetic circuit is laminated and consists of thin sheets stacked one on the other. These sheets are a ferromagnetic alloy of iron and silicon, which can work at a maximal induction up to 2.35T when the frequency is low. The thickness of a metal sheet is 0.33mm, 0.5mm or 0.65mm. The safety transformer respects the insulation temperature class E, that means the primary and secondary windings can withstand temperatures up to 120°C.

The primary and secondary copper windings are wound around the frame surrounding the central core and each part of lamination is assembled on this frame as shown in Fig 3.1. The primary winding is connected to the source and the secondary one is connected to the receiver.

Models will be presented in details in the following sections and used with the former methods for fast and heavy models in the optimization problem.

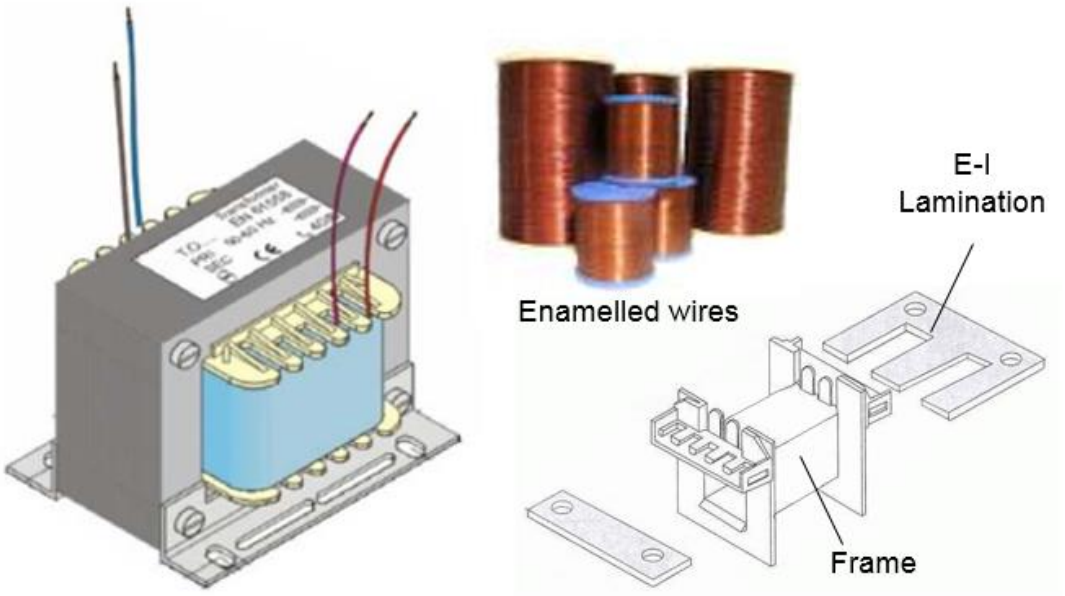


Fig. 3.1. Structure of the safety transformer (Tran et al., 2007)

1 Models

Two models of the transformer are used in this manuscript: one is an analytic model based on equivalent circuit and the other is a finite element model. First, AM will be used to test the methods adapted to fast models in order to verify our conclusion in the first chapter. Then, only the methods adapted to heavy models will be tested on the FEM because it is too complex and time consuming to be used with methods for fast models. Finally, a conclusion including the optimization results for AM and FEM will be drawn at the end of this chapter.

1.1 Analytic model

The physical phenomena within the transformer are thermal, electric and magnetic. They are expressed in equations that are ranked using specific algorithms.

There are some assumptions for AM:

1. The induction in the iron core is uniformly distributed.
2. No voltage drop due to the magnetizing current.
3. The magnetic field in coils is vertical.
4. The insulator between the core and the coils is in perfect contact with both parts.
5. There is no thermal contact between the exterior coil and the magnetic circuit.
6. There is no thermal exchange with the air trapped between the coils and the iron.
7. There is no convection on the upper and lower sides of the coil.
8. There is no temperature gradient in the copper and the iron.
9. All surfaces have the same convection coefficient.

The first two assumptions are based on Kapp's hypothesis (Seguier & Notelet, 1994). Without these two assumptions, the voltage drops caused by the passage of the magnetizing current in primary resistance will be taken into account, then the magneto-electric model is shown as in Fig 3.2.

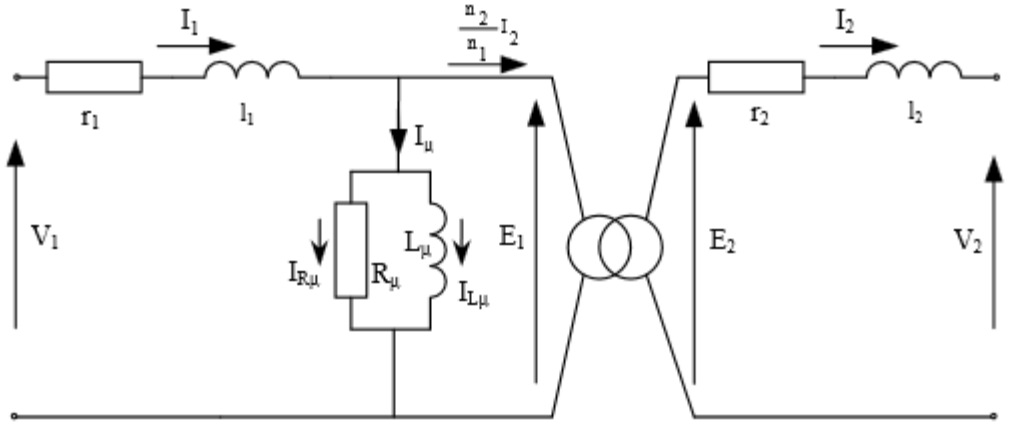


Fig. 3.2. Transform equivalent of the magneto electric model (Tran, 2009)

n_i is the number of turns for the coil, E_i represents the electromotive force, l_i is the leakage inductance, V_i is the voltage, r_i denotes the resistor. Index $i = 1,2$ means that the parameter is related to the primary or secondary coil. R_μ, L_μ are magnetizing resistor and inductance, I_1, I_2, I_μ are the primary, secondary and magnetizing currents.

The maximal induction B_m can be calculated by:

$$B_m = \frac{1}{4\pi} \cdot E_1 \frac{\sqrt{2}}{n_1 \cdot a \cdot d \cdot f} \tag{3.1}$$

where f is the frequency, a is the width of the column of E-I lamination and d is the thickness of the frame shown in Fig 3.1.

This model leads to an implicit system of 21 equations, in order to simplify the system, Kapp's hypothesis are introduced and the voltage drop is neglected.

Then the magneto electric model can be transformed equivalently to the electrical diagram as follows:

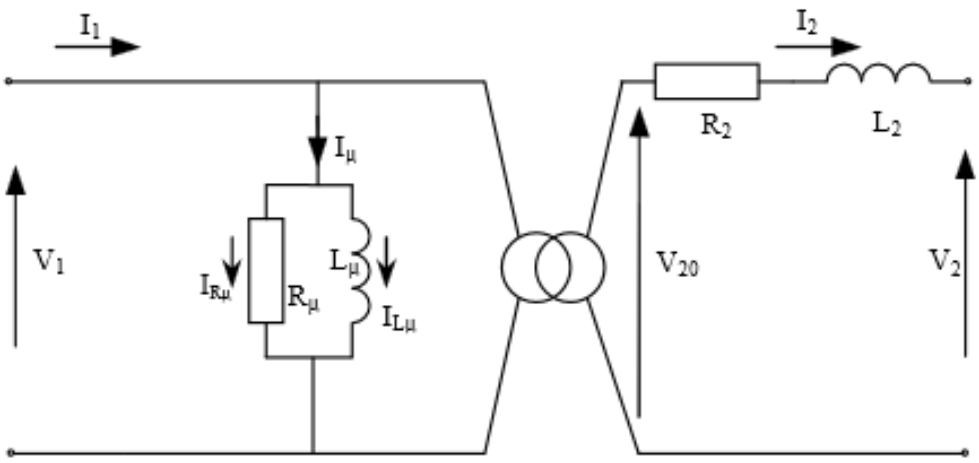


Fig. 3.3. Transform equivalent of the magneto electric model with Kapp's hypothesis (Tran, 2009)

V_{20} represents the voltage at no load, R_2, L_2 are the total magnetizing resistor and inductance which can be calculated by:

$$R_2 = r_2 + \left(\frac{n_2}{n_1}\right)^2 \cdot r_1 \tag{3.2}$$

$$L_2 = \frac{\varphi_2}{I_2} + \left(\frac{n_2}{n_1}\right)^2 \cdot \frac{\varphi_1}{I_1} \tag{3.3}$$

where φ_1, φ_2 are the total leakage fluxes of the two coils. Now the maximal induction B_m can be written as:

$$B_m = \frac{1}{4\pi} \cdot V_1 \frac{\sqrt{2}}{n_1 \cdot a \cdot d \cdot f} \tag{3.4}$$

This magneto electric model contains an implicit system of 8 multiphysical equations and the results from (Tran, 2009) shows that the data of this model are very close to those of the model without Kapp's hypothesis. So that this simpler model will be used as the AM of transformer since it does not lose many accuracy.

The initial nodal thermal model can be derived as shown in the left of Fig. 3.4, then with the $Y - \Delta$ transform which is also known as Kennely transform, it can be established equivalently as in the right part of Fig 3.4.

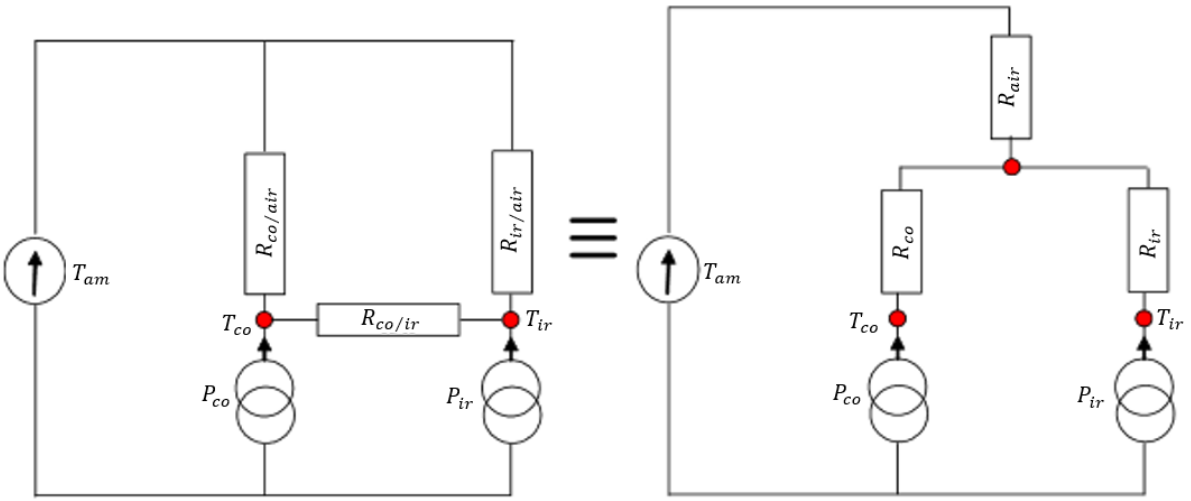


Fig. 3.4. Transform equivalent of the nodal thermal model (Tran, 2009)

T_{co}, T_{ir} and T_{am} represent the copper, iron and ambient temperature respectively, $R_{co/air}$, $R_{ir/air}$ and $R_{co/ir}$ are the thermal resistances between copper and air, iron and air, copper and iron respectively, P_{co} and P_{ir} denote the copper and iron losses, R_{co}, R_{ir} and R_{air} are equivalent thermal resistances of copper, iron and air.

With this transformed model, the copper temperature of coils and the iron temperature of magnetic circuit can be de deduced:

$$T_{co} = T_{am} + \frac{R_{ir/air} \cdot P_{co} + R_{ir/air} \cdot P_{ir} + R_{co/ir} \cdot P_{co}}{R_{co/air} + R_{ir/air} + R_{co/ir}} \cdot R_{co/air} \tag{3.5}$$

$$T_{ir} = T_{am} + \frac{R_{co/air} \cdot P_{co} + R_{co/air} \cdot P_{ir} + R_{co/ir} \cdot P_{ir}}{R_{co/air} + R_{ir/air} + R_{co/ir}} \cdot R_{ir/air} \tag{3.6}$$

The system of analytical equations is ranked using specific algorithms (Duff & Reid, 1978). The ranking of the system of non-linear equations that models magnetic, electric and thermal phenomena allows detecting the implicit system in order to assure the robustness of analytical model. Equations are detailed and explained in (Tran, 2009).

1.2 Finite element model

In order to build a more accurate model, FEM is considered. Thermal and magnetic phenomena are modeled by using 3D FEA on the eighth of transformer due to symmetries. There are about 43,000 nodes and 290,000 edges in the model. Fig 3.5 shows the mesh in the magnetic circuit, the insulating, the air gap, the frame and the opposing direction of currents in the primary and secondary windings that create flux in the gap between the coils (leakage flux).

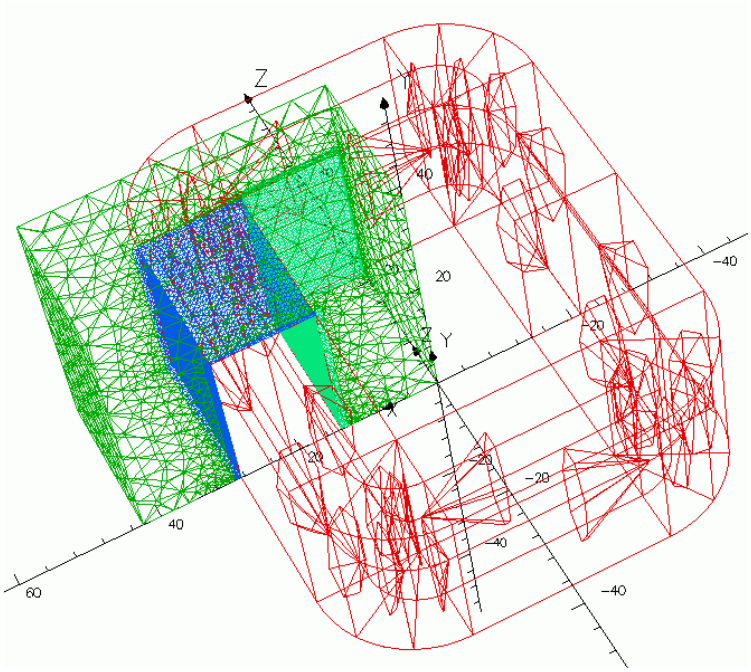


Fig. 3.5. Mesh of 3D FEA for the transformer

For the electromagnetic modeling, all magnetic and electric quantities are assumed sinusoidal. Full-load and no-load simulations are used to compute all the characteristics. The iron losses are computed with the Steinmetz formula and the leakage inductances are calculated with the magnetic co-energy. The core magnetic nonlinearity is taken into account. The inductance in the magnetic circuit can be observed on the eighth of transformer shown in the figure below. The flow created by the coils is mainly in the iron but there is also a leakage flux in the air and the coils, this leakage field is maximum between the two coils.

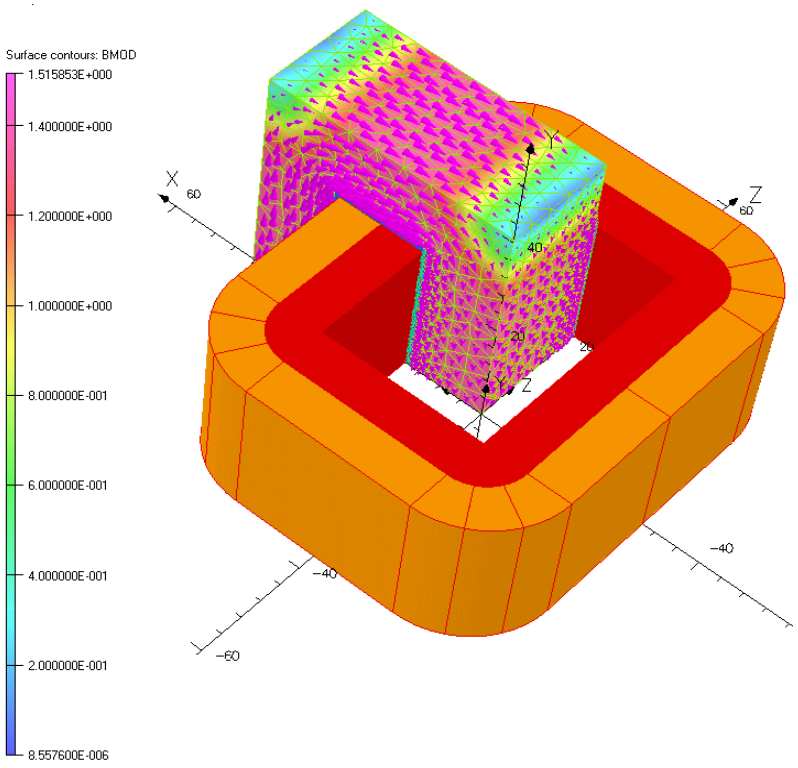


Fig. 3.6. 3D Finite element magnetic simulation (Tran, Brisset & Brochet., 2007)

In the thermal modeling, some assumptions are considered:

1. The insulator between the core and the coils is in perfect contact with both parts;
2. There is no thermal contact between the exterior coil and the magnetic circuit;
3. There is no thermal exchange with the air trapped between the coils and the iron;
4. There is no convection on the upper and lower sides of the coil;
5. There is no temperature gradient in the copper and the iron, and all surfaces have the same convection coefficient.

Thermal simulation of finite element is shown as the Fig. 3.7, it can be seen that the temperatures in the magnetic circuit are not uniform, which is different with AM. It appears that copper has higher temperature than iron and the parts under the iron is the hottest.

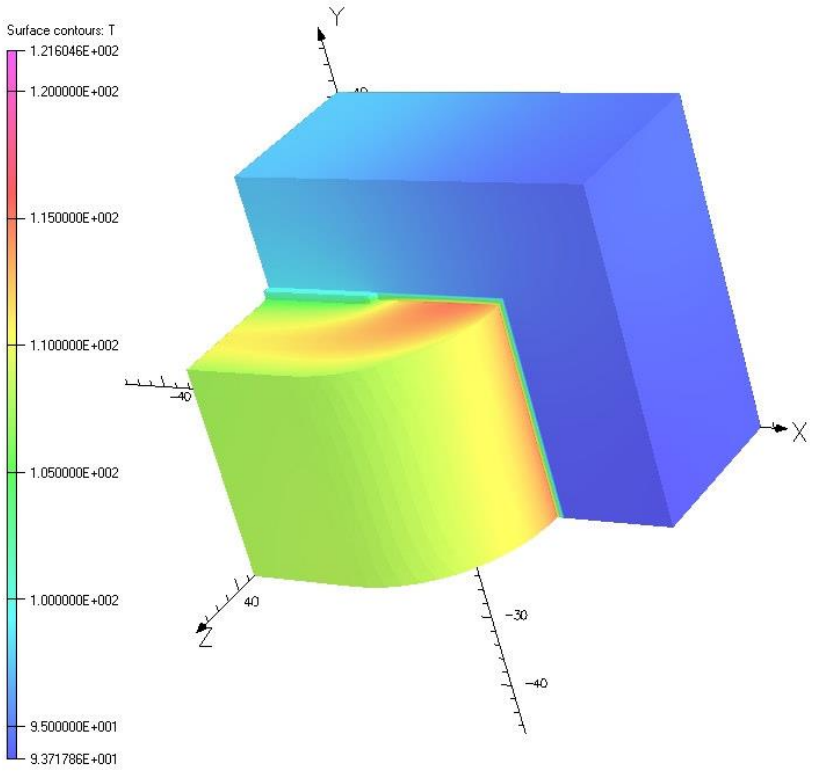


Fig. 3.7. 3D Finite element thermal simulation (Tran, 2009)

A magneto-thermal weak coupling is also considered in 3D FEM, The computational time is equal to 10 minutes on a single core of an Intel Xeon CPU E5-2690 at 2.60 GHz (with the same platform, AM only takes less than a second for one evaluation). The copper and iron losses are computed with the magnetic AC solver and introduced as heat sources in the thermal static solver. The copper temperature is used to compute the resistances of coils introduced in the magnetic solver and this loop continues until change in temperatures is less than 0.1°C. Both solvers use the same mesh and are included in Opera3D software.

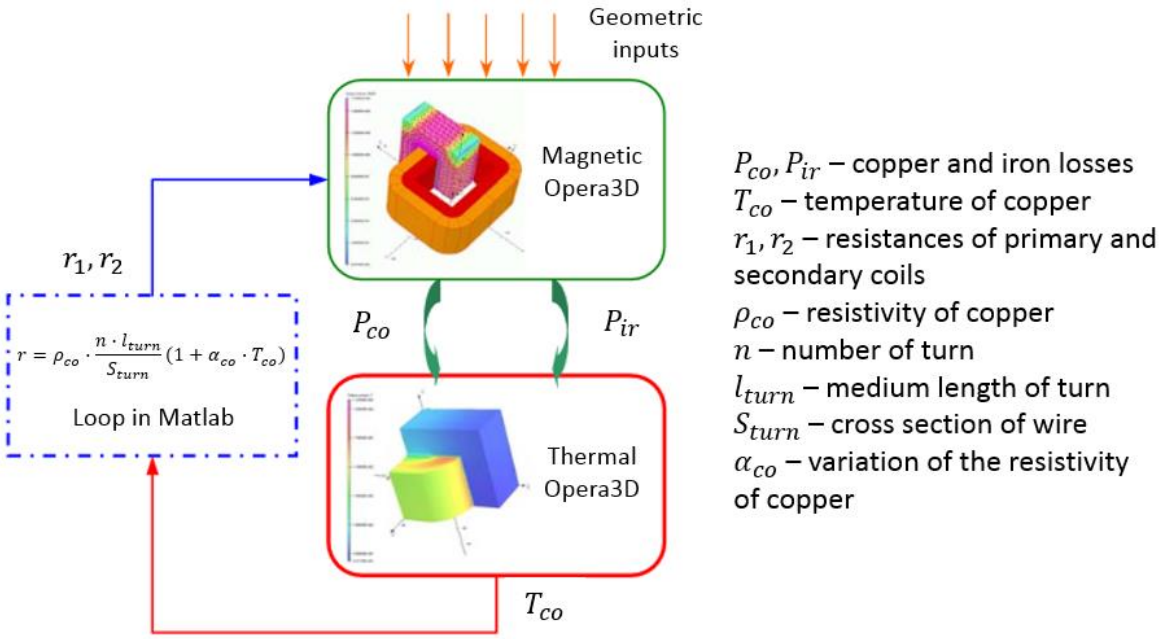


Fig. 3.8. 3D FEA magneto-thermal weak coupling (Tran, Brisset & Brochet., 2007)

1.3 Comparison AM and FEM

Before applying the methods to the optimization problem, a series of geometrical configuration is selected to compare the two former models: $a = 15\text{mm}, b = 45\text{mm}, c = 30\text{mm}, d = 30\text{mm}, n_1 = 800, S_1 = 0.422\text{mm}^2, S_2 = 3.672\text{mm}^2$, where a, b, c are the parameters for the shape of lamination, d is the thickness of the frame, S_1, S_2 are the primary and secondary section of conductors, and n_1 is the number of primary turn.

Besides, other configurations necessary are selected as: power 192 (VA), primary voltage $V_1 = 230\ (\text{V})$, secondary voltage $V_2 = 24\ (\text{V})$, power factor $f_p = 0.8$, ambient temperature $T_{am} = 40\ (°\text{C})$, coefficient of heat transfer by convection $h = 10\ (\text{W} \cdot \text{m}^{-2} \cdot \text{K}^{-1})$, thermal conductivity of insulating material $0.15(\text{W} \cdot \text{m}^{-1} \cdot \text{K}^{-1})$, the resistivity of copper $\rho_{co} = 1.72 \times 10^{-8}\ (\Omega \cdot \text{m})$, variation of the resistivity of copper $\alpha_{co} = 3.8 \times 10^{-8}\ (\text{K}^{-1})$.

The results of both AM and FEM are presented in Table 3.1. The relative error is calculated by the percentage of difference between AM and FEM values divided by FEM ones. From the results, it can be seen that the 3D FEM is more accurate but it requires a great amount of time because of the magneto-thermal coupling. Besides, some results of the two models are rather close but there are also some others whose difference cannot be ignored, for example the relative error of copper temperature is -14.484%, and that is due to the temperatures in the winding and in the magnetic circuit are not uniform for 3D FEM which differs from the assumptions of AM.

Table 3.1. Comparison of analytical and 3D FE models (Tran, Brisset & Brochet, 2007)

Parameters			AM	FEM	Relative error
Primary resistor	r_1	Ω	7.763	8.088	-4.018%
Secondary resistor	r_2	Ω	0.158	0.166	-4.819%
Primary leakage inductance	l_1	mH	17.040	16.980	0.353%
Second leakage inductance	l_2	mH	0.280	0.245	14.286%
Number of secondary turn	n_2	--	162.609	164.529	-1.167%
Primary current	I_1	A	0.981	0.947	3.590%
Magnetizing current	I_μ	mA	83.817	67.553	24.076%
Magnetizing inductance	L_μ	H	8.846	10.214	-13.393%
Maximal induction	B_m	T	1.438	1.281	12.256%
Copper loss	P_{co}	W	16.623	17.573	-5.406%
Iron loss	P_{ir}	W	3.048	2.490	22.410%
Copper temperature	T_{co}	ºC	107.40	125.59	-14.484%
Iron temperature	T_{ir}	ºC	96.18	94.81	1.445%
Efficiency	η	%	88.6	88.13	0.533%
Voltage drop	ΔV_2	V	2.424	2.736	-11.404%
Computational time	t	S	0.5	6942.6	--

As the windings can withstand temperatures up to 120 ℃, so that the copper temperature is used in the constraint $T_{co} \leq 120$, the underestimate of AM will enlarge the security domain and leads to an unfeasible solution. Additionally, the magnetizing current I_μ, voltage drop ΔV_2 whose values are not accurate enough in AM may also influence other electromagnetic constraints.

Moreover, in manufacturing, our purpose is to produce machines that meet the performance requirements with the least amount of material. Thus the total mass of the material m_{tot} is an important variable we concerned, which is the sum of iron mass m_{ir} and copper mass m_{co}. They can be calculated by the following expressions:

$$m_{ir} = 4a \cdot b \cdot (2a + b + c) \cdot \rho_{ir} \tag{3.7}$$

$$m_{co} = \left[\left(2d + 4a + \frac{\pi c}{2}\right) n_1 S_1 + \left(2d + 4a + \frac{3\pi c}{2}\right) n_2 S_2\right] \cdot \rho_{co} \tag{3.8}$$

where ρ_{ir} and ρ_{co} are the densities of iron and copper. For m_{ir}, it depends only on the inputs directly, but for m_{co}, it depends on intermediate variable n_2 also. Thus, the imprecise estimate of n_2 calculated by AM will lead to an imprecise solution of total mass.

Although FEM is more precise, it is impossible to use it for an optimization with methods for fast models due to the huge time consuming, thus methods for heavy models come in handy.

2 Optimization problem

The optimization design problem for transformer contains 7 design variables. There are three parameters (a, b, c) for the shape of the lamination, one for the frame (d), two for the section of conductors (S_1, S_2), and one for the number of primary turn (n_1) as shown in Fig 3.9.

Their lower and upper bounds are presented in Table 3.2.

Table 3.2. Lower and upper bounds for seven design variables

Design variables	$a(mm)$	$b(mm)$	$c(mm)$	$d(mm)$	n_1	$S_1(mm^2)$	$S_2(mm^2)$
Lower bound	3	14	6	10	200	0.15	0.15
Upper bound	30	95	40	80	1200	19	19

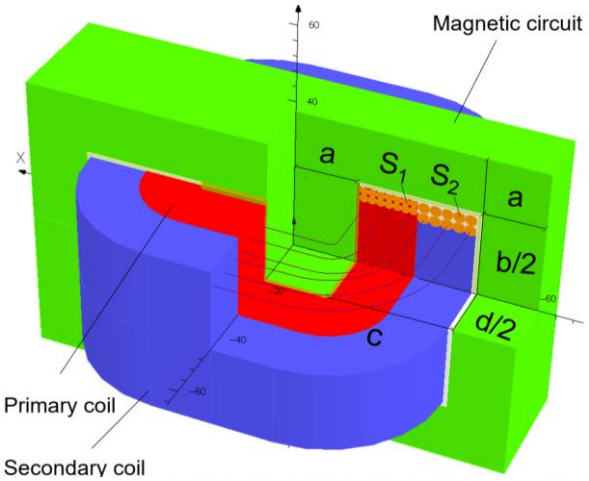


Fig. 3.9. The design variables of the transformer

There are also 7 inequality constraints in this problem. The copper and iron temperatures T_{co}, T_{ir} should be less than 120℃ and 100℃, respectively. The efficiency η should be greater than 80%. The magnetizing current I_μ/I_1 and drop voltage $\Delta V_2/V_2$ should be less than10%. All these constraints are computed with Finite Element Model (FEM) or Analytic Model (AM) model. Finally, the filling factors of both coils f_1, f_2 should be lower than 0.5.

The goal is to minimize the mass m_{tot} of iron and copper materials. Thus, the optimization problem is expressed as:

$$\begin{aligned} &\min m_{tot}(a, b, c, d, n_1, S_1, S_2) \\ &s.t.\ \ \boldsymbol{g}(a, b, c, d, n_1, S_1, S_2) \leq 0 \end{aligned} \tag{3.9}$$

where

$$\boldsymbol{g} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \\ g_6 \\ g_7 \end{bmatrix} = \begin{bmatrix} T_{co} - 120 \\ T_{ir} - 100 \\ \frac{I_\mu}{I_1} - 0.1 \\ \frac{\Delta V_2}{V_2} - 0.1 \\ f_1 - 0.5 \\ f_2 - 0.5 \\ 0.8 - \eta \end{bmatrix} \tag{3.10}$$

Other configurations needed are fixed with the same values as presented in section 1.3 during the optimization design problem.

To verify the feasibility and efficiency of the aforementioned methods in this manuscript, all the design variables are considered as following the Gaussian distribution while the standard deviation is 1% of the lower bounds of each variable. As an industrial electromagnetic device, in order to increase the quality, the target probability of failure is set to be P_t =0.13% that means reliability index is $\beta_t = 3$, the confidence level k is also set to be 3.

3 Comparison of methods for fast models

From the previous section, it can be noticed that AM of the transformer is fast to evaluate, it is natural to wonder whether the optimization problem can be solved with AM by the methods for fast models or not. To verify this question and to make a more general conclusion for the methods, AM of the transformer will be tested in this manuscript with methods mentioned in Chapter 1. Following parts will show the results of optimization given by different categories of methods. For the purpose of comparison, results of DDO will be presented in each type of category.

3.1 WCO methods

The problem of WCO for transformer can be written as:

$$\begin{gathered} \min_{\boldsymbol{d}} m_{tot_w}(\boldsymbol{d}) \\ s.t.\ g_{w_i}(\boldsymbol{d}) \leq 0 \\ \boldsymbol{d^L} \leq \boldsymbol{d} \leq \boldsymbol{d^U} \end{gathered} \tag{3.11}$$

with

$$\begin{aligned} m_{tot_w}(\boldsymbol{d}) &= \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} m_{tot}(\boldsymbol{\xi}) \\ g_{w_i}(\boldsymbol{d}) &= \max_{\boldsymbol{\xi} \in U(\boldsymbol{d})} g_i(\boldsymbol{\xi}) \end{aligned} \tag{3.12}$$

where m_{tot_w} presents the worst-case of objective m_{tot}, $\boldsymbol{d} = [a, b, c, d, n_1, S_1, S_2]^{\boldsymbol{T}}$ is the vector of input variables, $U(\boldsymbol{d}) = \{\boldsymbol{\xi} \in \mathbb{R}^7 | \boldsymbol{d} - 3\boldsymbol{\sigma} \leq \boldsymbol{\xi} \leq \boldsymbol{d} + 3\boldsymbol{\sigma}\}$ is the uncertainty set

around the current design point $\boldsymbol{d}$ and the standard deviation of inputs $\boldsymbol{\sigma} = 0.01\boldsymbol{d^L}$, $i = 1, \dots, 7$.

Traditional WCO method (the one that uses inner optimization loop to calculate the worst-case for each point) and also WWCO, GWCO are all tested here. However in the Table 3.3 there are only results of WWCO and GWCO, traditional WCO is too complex and time consuming for this complicated model, the number of evaluations is out of our limit that is 10^5 whereas it could not find the optimum. It is because that for each objective function and constraint, it will introduce a sub-optimization process to calculate the worst-case of current design point. Thus for this transformer model which has one objective and 7 constraints, it needs 8 sub-optimization process. Moreover due to the multimodal and highly constrained characteristics of this optimization problem, these sub-optimizations may lead to the failure of finding out the global optimum correctly if no multi-start is used. In fact, even WWCO that evaluates only $(2p + m + 1)$ times the model (p denotes the dimension of design variables and m is the number of constraints) for each design point requires more than 3×10^4 evaluations in total.

Table 3.3. Results of transformer optimization with different WCO methods

Variables	DDO	WWCO	GWCO	
$a(mm)$	13.004	13.211	12.867	
$b(mm)$	50.1	54.025	64.149	
$c(mm)$	16.537	16.887	22.991	
$d(mm)$	43.05	41.453	43.603	
n_1	639.76	670.57	670.15	
$s_1(mm^2)$	0.3238	0.3264	0.3431	
$s_2(mm^2)$	2.9026	2.9532	3.3165	
$\mu_{m_{tot}}(kg)$	2.3112	2.3950	2.8701	
$\sigma_{m_{tot}}(kg)$	0.0079	0.0082	0.0093	
Convergence (with 100 runs)	12%	78%	66%	
m_{tot_w}	2.3648	2.4499	3.0899	
g_{w_1}	-0.0779	-0.0950	-0.0977	-0.2101
g_{w_2}	0.0152	0	0	-0.1137
g_{w_3}	-0.2872	-0.2692	-0.0656	-0.2530
g_{w_4}	0.2185	0	0	-0.0210
g_{w_5}	0.0436	-0.0001	0	-0.3533
g_{w_6}	0.0322	0	-0.2791	-0.3076
g_{w_7}	-0.1172	-0.1154	-0.0841	-0.1140
Evaluations	242	36127	11218	

The lines m_{tot_w} and g_{w_i} $(i = 1, \dots, 7)$ present the worst-case of objective function and constraints. It can be seen that DDO has smaller values of $\mu_{m_{tot}}$, $\sigma_{m_{tot}}$ and even m_{tot_w}, but

there are four worst-case constraints violated. As a contrast, other two methods of WCO can respect all the worst-case constraints.

We focus on the two methods which can converge successfully within the limitation of evaluations number. WWCO needs almost 3 times more evaluations than GWCO. Comparing with the results in Chapter 1, it seems that GWCO is much faster as it only needs $(p+1)$ times of evaluation for each design point while WWCO needs $(2p+m+1)$ times. And it is obverse that with the increase of the constraint and input variable numbers, this advantage of GWCO becomes more and more pronounced.

However, the results of GWCO are not advisable as both the mean $\mu_{m_{tot}}$ and the standard deviation of total mass $\sigma_{m_{tot}}$ are larger than WWCO's. The left column of g_{w_i} for GWCO is calculated by the method itself, and right one is obtained by WCO, from these results it can be noticed that in fact the optimum found by GWCO is located on none of the worst-case limit-states. This method underestimate the worst-cases.

This is the same as we have concluded in Chapter 1, traditional WCO needs the most number of evaluations and this time it is not affordable with a more complicated model as it fails to converge within the limitation of evaluations number. GWCO is the fastest among those three methods but it could not achieve the minimum performance. That is due to its principle of using an approximation to replace the exact worst case. Although sometimes WWCO could not find the exact worst case either and substitutes it with a vertex but, at least, the substitution is in the uncertainty set. On the contrary sometimes the approximation of GWCO may lie out of the set. So among them, WWCO is the most balanced and suitable method.

3.2 RDO methods

A robust problem of transformer involves the means and standard deviations of objective m_{tot} and constraints:

$$\begin{gathered}\min_{\boldsymbol{d}} f_r\left(\mu_{m_{tot}}(\boldsymbol{d}), \sigma_{m_{tot}}(\boldsymbol{d})\right) \\ s.t.\ \ g_{r_{\mathrm{i}}}(\boldsymbol{d}) \leq 0 \\ \boldsymbol{d}^L + 3\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - 3\boldsymbol{\sigma}\end{gathered} \tag{3.13}$$

where

$$g_{r_{\mathrm{i}}}(\boldsymbol{d}) = \mu_{g_i}(\boldsymbol{d}) + 3\sigma_{g_i}(\boldsymbol{d}) \tag{3.14}$$

$\mu_{g_i}(\boldsymbol{d})$ and $\sigma_{g_i}(\boldsymbol{d})$ are the mean and standard deviation of the i-th original constraints presented in Equation (3.10), $i = 1, \dots 7$, and $\mu_{m_{tot}}(\boldsymbol{d}), \sigma_{m_{tot}}(\boldsymbol{d})$ are the moments of objective m_{tot}.

In Chapter 1, section 2.2, 9 different formulations of RDO have been tested with a simple mathematic example, and after the discussion, it was concluded that Formulation

1 to 3 could find only one solution as their weight parameters for mean and standard deviation of initial objective function are fixed. Formulation 5 to 7 can reach different optimal points on account of different values of weight parameters. However, they depend on the function type of the two objectives, if it is convex, a whole Pareto front can be obtained, but if it is concave, only the bounds of this front can be reached. The three other formulations including ϵ-constraint and multi-objective algorithms perform well in the application of mathematic example. Now, all these nine methods are tested with this more complicated model of transformer to verify the former conclusions.

Table 3.4, 3.5 and Fig 3.10 to 3.12 present the results of those multi-objective methods. Notice that before running optimization process of Formulation 4, the lower and upper bounds of $\sigma_{m_{tot}}$ are needed. This information is calculated by minimizing only $\sigma_{m_{tot}}$ and $\mu_{m_{tot}}$ respectively and the values are $\sigma_{fmin} = 8.055 \times 10^{-3}, \sigma_{fmax} = 8.086 \times 10^{-3}$.

Table 3.4. Results of transformer optimization with first three RDO formulations

Variables	DDO	Formulation 1	Formulation 2	Formulation 3
$a(mm)$	13.004	13.069	13.059	13.405
$b(mm)$	50.1	52.177	52.065	45.727
$c(mm)$	16.537	16.758	16.781	17.141
$d(mm)$	43.05	42.232	42.309	45.036
n_1	639.76	657.36	656.60	603.31
$s_1(mm^2)$	0.3238	0.3256	0.3257	0.3177
$s_2(mm^2)$	2.9026	2.9294	2.9303	2.8606
$\mu_{m_{tot}}(kg)$	2.3112	2.3524	2.3524	2.3572
$\sigma_{m_{tot}}(kg)$	0.0079	0.0081	0.0081	0.0081
Convergence (with 100 runs)	12%	68%	75%	35%
g_{r_1}	-0.0863	-0.0939	-0.0939	-0.0850
g_{r_2}	0.0065	0	0	0
g_{r_3}	-0.2990	-0.2877	-0.2877	-0.2921
g_{r_4}	0.1055	0	0	0
g_{r_5}	0.0218	0	0	0
g_{r_6}	0.0171	0	0	0
g_{r_7}	-0.1183	-0.1171	-0.1172	-0.1182
Evaluations	242	5943	5723	4651

We first concentrate on the first three formulations which can only obtain just one solution. From Table 3.4, lines g_{r_i} $(i = 1, \dots, 7)$ present the robust constraints, in this situation, results of DDO no longer satisfy the constraints. It can be seen that Formulation 1 and 2 achieve almost the same results while Formulation 3 could not converge to the global minimum. This situation is similar to in Chapter 1 and the reason is that the order

of magnitude difference between $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ is great so even though Formulation 2 adds an extra term $3\sigma_f(\boldsymbol{d})$ in the robust objective compared with Formulation 1, it could not make influence to the result and both of them lead to the extreme of the Pareto front. Then for Formulation 3, the scaled values eliminate the gap of order of magnitude, so that it converges to another point.

And for Formulation 4, the ϵ-constraint method, as the variance of σ_f is not very large, we divide the interval $[\sigma_{fmin}, \sigma_{fmax}]$ into 20 sections. Fig 3.10 shows the Pareto front found by Formulation 4.

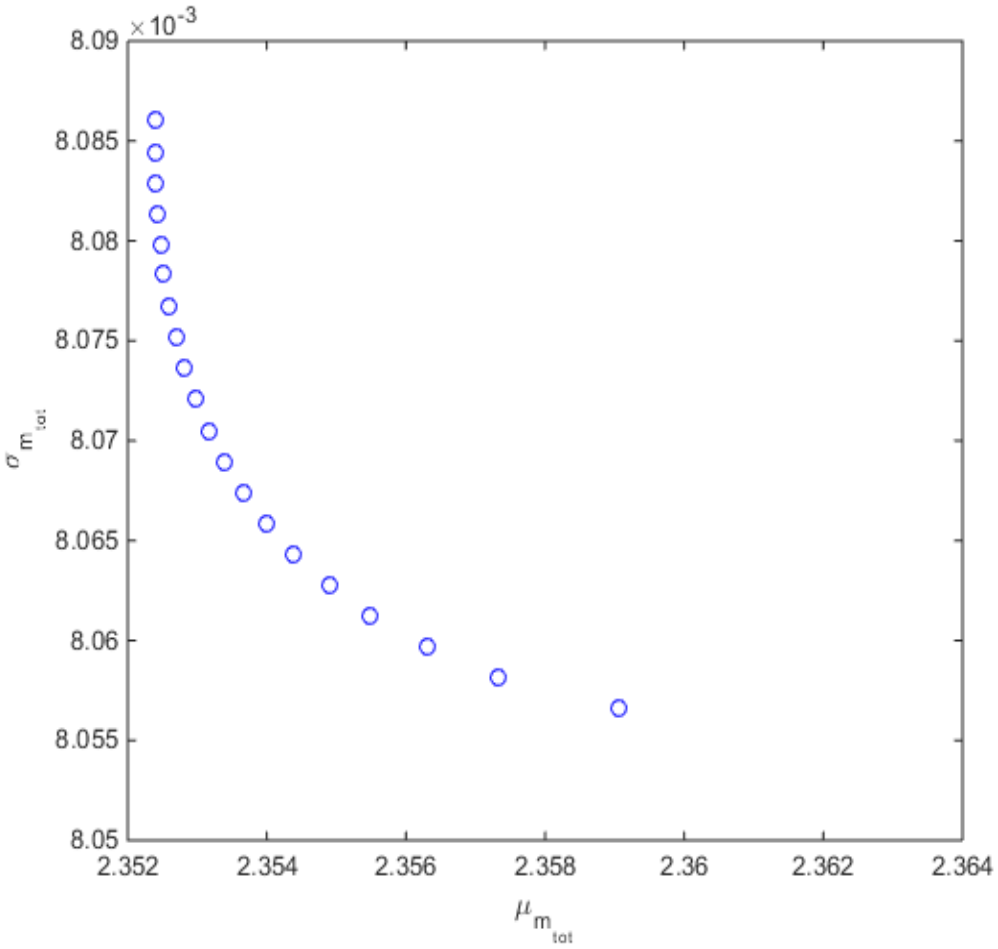


Fig. 3.10. Results of transformer with Formulation 4

Formulation 5, 6 and 7 are the methods with variable weights, the only difference is that Formulation 5 uses square information, Formulation 7 uses scaled values and Formulation 6 uses just σ_f and μ_f. However, as there exist a large difference in order of magnitude between $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ in this problem, the square in Formulation 5 increase this difference and could only find one solution no matter how to change the weight. Its solution is shown in following table.

Table 3.5. Results of transformer optimization with RDO Formulation 5

Variables	$a(mm)$	$b(mm)$	$c(mm)$	$d(mm)$	n_1	$s_1(mm^2)$	$s_2(mm^2)$	$\mu_{m_{tot}}(kg)$	Convergence
Values	13.071	52.193	16.758	42.214	657.47	0.3256	2.9295	2.3524	69%
Variables	g_{r1}	g_{r2}	g_{r3}	g_{r4}	g_{r1}	g_{r1}	g_{r1}	$\sigma_{m_{tot}}(kg)$	Evaluations
Values	-0.0969	0	-0.2741	0	0	0	-0.1167	0.0081	6053

Results of Formulation 6 and 7 are in Fig. 3.11 and 3.12, it can be seen that this time the function of $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ is convex so that these two formulations can obtain a Pareto front. Formulation 7 is more balanced owing to its scaled values.

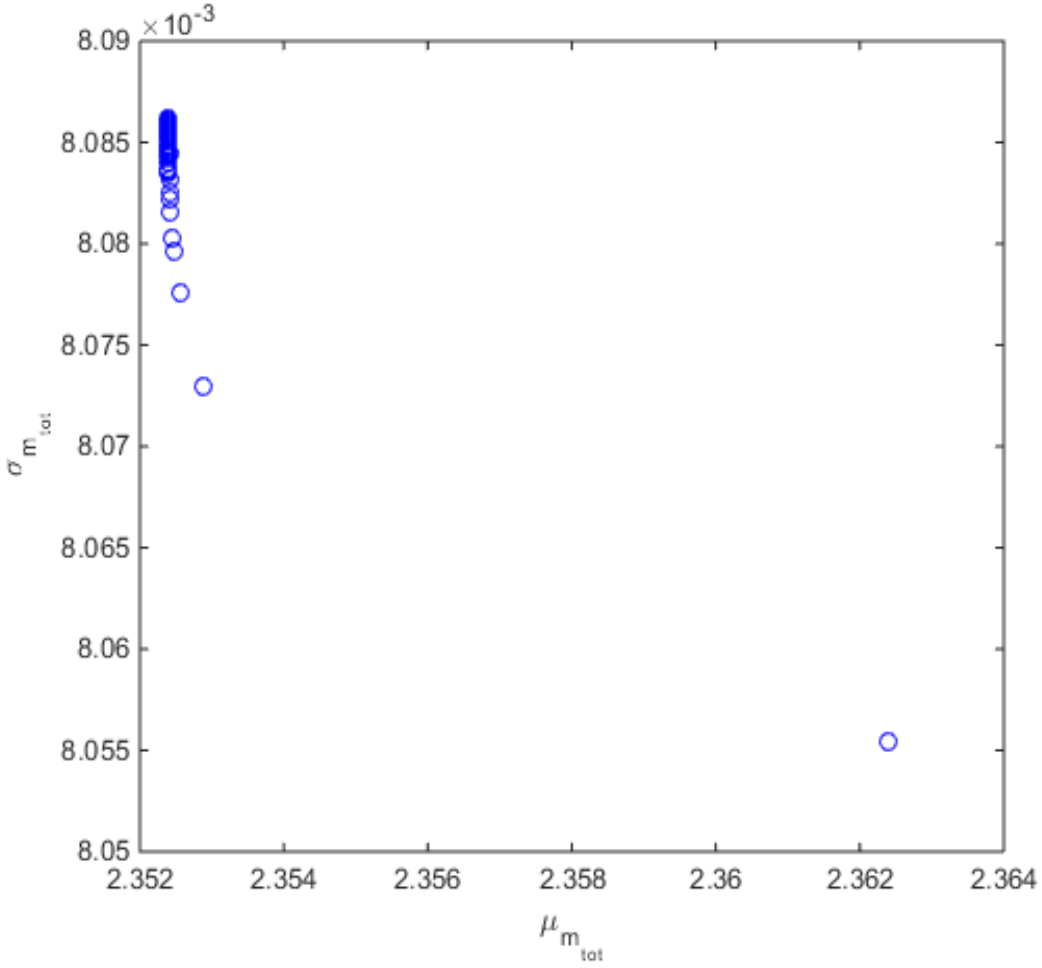


Fig. 3.11. Results of transformer with Formulation 6

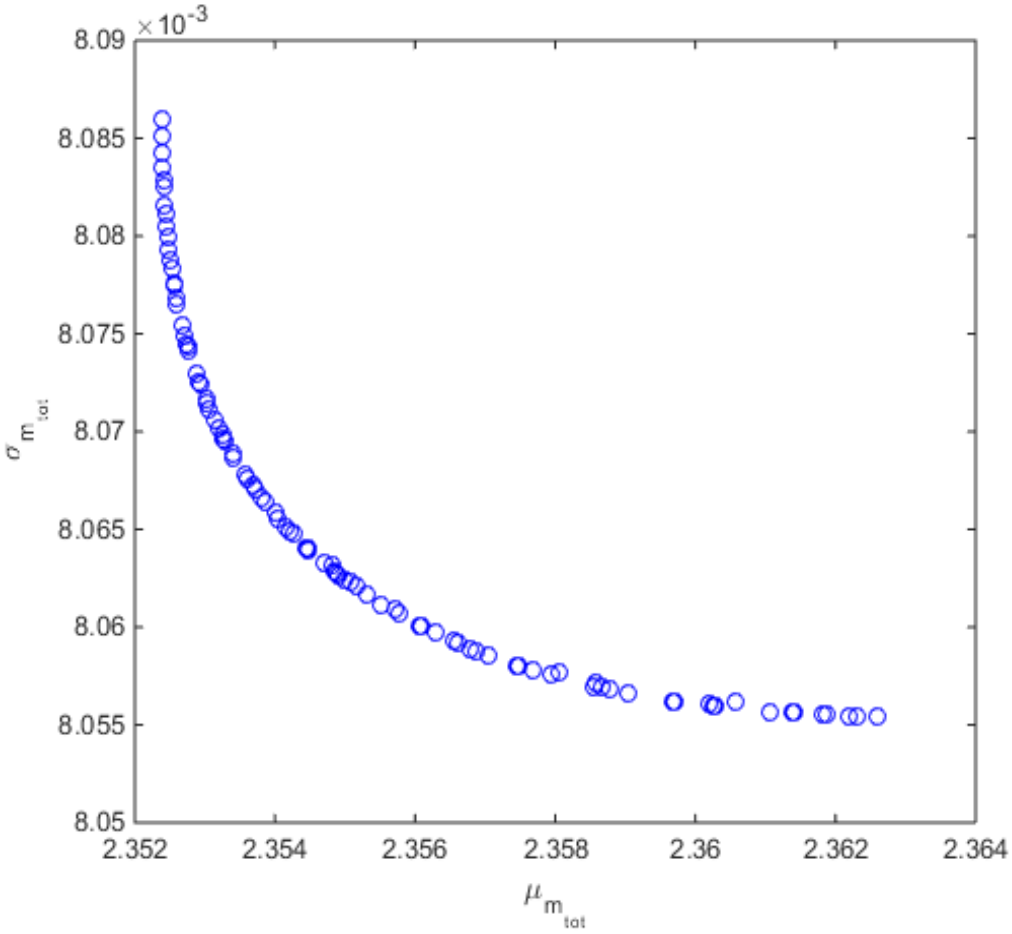


Fig. 3.12. Results of transformer with Formulation 7

Then for multi-objective methods as Formulation 8 and 9, we fix the goal for goal-attain algorithm as the robust minimum of $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ which can be obtained by minimizing them with robust constraints, so that is $[2.3524; 8.055 \times 10^{-3}]$. However, neither goal-attain algorithm nor NSGA2 could achieve the minimum within the limitation of function evaluation number, even if we change the goal to a non-attainable value. That is a consequence of highly constrained and multimodal.

From all above results, the conclusion for this fast electromagnetic analytical model has some distinctions with the simple mathematic model in Chapter 1. For the simple mathematic model, the most efficient formulations are the formulation with ϵ-constraint method (Formulation 4) and multi-objective algorithms (Formulation 8, 9). However, for this analytical electromagnetic model, it can be seen that Formulation 4 and 7 perform the best while multi-objective algorithms failed to converge.

Therefore, combine the results of both examples of transformer and mathematic model, we can conclude that multi-objective methods are always better than mono-objective ones. However, the algorithms used for solving the multi-objective methods may result in different performance.

ϵ-constraint method is the most universal but it needs more information like the lower and upper bounds of standard deviation of objective function. Moreover, how to divide the interval $[\sigma_{fmin}, \sigma_{fmax}]$ is another problem, if the sub-sections are too large, the method may fail to obtain a good Pareto front, but if they are too small, it will take a lot of time.

Scaled values with weight parameters depends on the Pareto front's curvature which is not available a priori, if it is concave, it could not obtain a complete Pareto front.

Multi-objective methods are not stable, sometimes they could not give a feasible result within limitation of evaluation.

3.3 RBDO methods

For RBDO, the optimization problem is transformed into:

$$\begin{gathered} \min_{\boldsymbol{d}} m_{tot}(\boldsymbol{d}) \\ s.t. \quad P_f \leq 0.13\% \\ \boldsymbol{d}^L + 3\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - 3\boldsymbol{\sigma} \end{gathered} \tag{3.15}$$

where

$$P_f = \begin{bmatrix} P(T_{co} > 120) \\ P(T_{ir} > 100) \\ P\left(\frac{I_\mu}{I_1} > 0.1\right) \\ P\left(\frac{\Delta V_2}{V_2} > 0.1\right) \\ P(f_1 > 0.5) \\ P(f_2 > 0.5) \\ P(\eta < 0.8) \end{bmatrix} \tag{3.16}$$

In Chapter 1 we focus on six methods belonging to the three different types of RBDO. Results have shown that with the simple mathematical example, sequential decoupled methods are the most suitable without losing any accuracy and other two types of methods cannot maintain the accuracy or require large amount of evaluations number.

This time all those six mentioned methods (RIA, PMA, AMA, SLA, SORA and SAP) are tested with transformer AM to check the former conclusion. However only three of them can converge, and it is coincidental that in each type of RBDO method there is one that does not work (RIA, AMA and SAP).

The results are shown in the following table. For comparison, there are two columns of probabilities of failures for each method, the left ones are calculated by the methods themselves and the right ones are calculated by Monte-Carlo simulation with 10^6 samples.

Table 3.6. Results of RBDO methods on the safety transformer

Method	DDO		PMA		SLA		SORA	
$a\ (mm)$	13.004		13.078		13.045		13.078	
$b\ (mm)$	50.1		52.274		51.613		52.278	
$c\ (mm)$	16.537		16.758		16.837		16.757	
$d\ (mm)$	43.05		42.182		42.603		42.180	
n_1	639.76		658.36		653.56		658.39	
$S_1\ (mm^2)$	0.3238		0.3256		0.3254		0.3256	
$S_2\ (mm^2)$	2.9026		2.9296		2.9273		2.9296	
$\mu_{M_{tot}}(kg)$	2.3112		2.3546		2.3534		2.3546	
$\sigma_{M_{tot}}(kg)$	0.0079		0.0081		0.0081		0.0081	
Convergence (with 100 runs)	12%		84%		72%		23%	
$P(T_{co} > 120℃)(\%)$	0	0	0	0	0	0	0	0
$P(T_{ir} > 100℃)(\%)$	50	50.658	0.13	0.1337	0.13	0.3754	0.13	0.1389
$P(I_{\mu}/I_1 > 0.1)(\%)$	0	0	0	0	0	0	0	0
$P(\Delta V_2/V_2 > 0.1)(\%)$	50	50.203	0.13	0.1356	0.13	0.1357	0.13	0.1370
$P(f_1 > 0.5)(\%)$	50	49.987	0.13	0.1323	0.13	0.1362	0.13	0.1388
$P(f_2 > 0.5)(\%)$	50	50.021	0.13	0.1415	0.13	0.1368	0.13	0.1382
$P(\eta < 0.8)(\%)$	0	0	0	0	0	0	0	0
Evaluations	242		30607		3024		2171	

Note that PMA can find an optimum that satisfies all the constraints but the number of evaluations of the model is high. For SLA, it has a smaller number of evaluations but there is one constraint violated. The reason is that SLA sacrifices the accuracy in order to reduce the number of evaluations, leading to a coarse computation of the probability of failure. The convergence rate of SORA is not as good as the other two but it has the smallest number of evaluations among them. It can be seen that this number is nearly 15 times less than PMA and even less than SLA. It is also owing to the coarse computation of SLA, more iterations are needed to find out a solution. Unfortunately, the rate of convergence for SORA is three of four times lower than other two. So that a multi-start process is required and the number of evaluations will increase consequently.

The conclusion from this transformer optimization has some similarities with mathematical example but also has some differences. The similarities are: The single-loop methods are always the most inaccurate method and double-loop methods have the highest number of evaluations, sequential decoupled methods are most efficient considering both the computing time and the accuracy.

The differences are: In simple mathematic example, SAP is faster than SORA and single-loop methods are generally faster than other two type of method. However, for more complicated examples like the transformer, some approaches like SAP fail to find a solution, probably because of the hard-constrained problem and the discontinuous

domains of security. So, it also indicates that not all the aforementioned approaches can handle complicated models. Moreover, the number of evaluations of SORA is smaller than of SLA for this optimization problem, SORA may be more effective for complex model. And Summarized from the two examples, SORA seems to be the most effective method among all the mentioned methods.

3.4 RBRDO methods

Also, all the mentioned 5 formulations of RBRDO in Chapter 1 are tested here with this electromagnetic AM to verify the previous conclusions. The formulation of RBRDO for transformer is written as:

$$\begin{gathered} \min_{\boldsymbol{d}} f_r\left(\mu_{m_{tot}}, \sigma_{m_{tot}}\right) \\ s.t. \quad P_f \leq 0.13\% \\ \boldsymbol{d}^L + 3\boldsymbol{\sigma} \leq \boldsymbol{d} \leq \boldsymbol{d}^U - 3\boldsymbol{\sigma} \end{gathered} \tag{3.17}$$

where the expression of P_f is the same as in Equation (3.16).

First we compare the mono-objective methods, thus the Formulation 4 and 5. The results are shown in Table 3.7.

Table 3.7. Results of RBRDO methods on the safety transformer

Design variables	DDO		Formulation 4		Formulation 5	
$a(mm)$	13.004		13.097		13.074	
$b(mm)$	50.1		52.449		52.236	
$c(mm)$	16.537		16.734		16.766	
$d(mm)$	43.05		42.033		42.211	
n_1	639.76		659.74		658.11	
$s_1(mm^2)$	0.3238		0.3256		3.2568	
$s_2(mm^2)$	2.9026		2.9290		2.9298	
$\mu_{m_{tot}}(kg)$	2.3112		2.3546		2.3546	
$\sigma_{m_{tot}}(kg)$	0.0079		0.0081		0.0081	
Convergence (with 100 runs)	12%		85%		78%	
$P(T_{co} > 120℃)(\%)$	0	0	0	0	0	0
$P(T_{ir} > 100℃)(\%)$	50	50.658	0.13	0.1449	0.13	0.1336
$P\left(I_{\mu}/I_1 > 0.1\right)(\%)$	0	0	0	0	0	0
$P(\Delta V_2/V_2 > 0.1)(\%)$	50	50.203	0.13	0.1375	0.13	0.1361
$P(f_1 > 0.5)(\%)$	50	49.987	0.13	0.1317	0.13	0.1362
$P(f_2 > 0.5)(\%)$	50	50.021	0.13	0.1351	0.13	0.1336
$P(\eta < 0.8)(\%)$	0	0	0	0	0	0
Evaluations	242		68846		76724	

Like RBDO, there are two columns of probabilities of failure, the left ones are calculated by the methods themselves, the right ones are obtained by Monte Carlo simulation with

10^6 samples. The results have the same regularity as with the simple mathematic example, we can notice that although the optima found by Formulation 4 and 5 have little differences, the values of $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ are the same as results of PMA and SORA in RBDO, it is the same as we have found in Chapter 1.

To compare other formulations, the target f_t for NTB is chosen as the optimum of DDO that is 2.3112. It should be noticed that as the probabilities of failure for several constraints are 50% for DDO optimum, it surpasses the target probability 0.13% greatly, so this target is non-attainable for RBRDO. μ_{f_0} , σ_{f_0} for scaling are fixed as $\mu_{f_0} = 2.7583, \sigma_{f_0} = 0.0135$. The weight parameter is changed, for example ω_1 from 0 to 1 with step 0.1 while ω_2 from 1 to 0 with negative step, the results are shown in following figures. It should be noticed that the scales of horizontal and vertical axis of Fig 3.15 are not the same as other two figures because that LTB finds solutions far from the minimum.

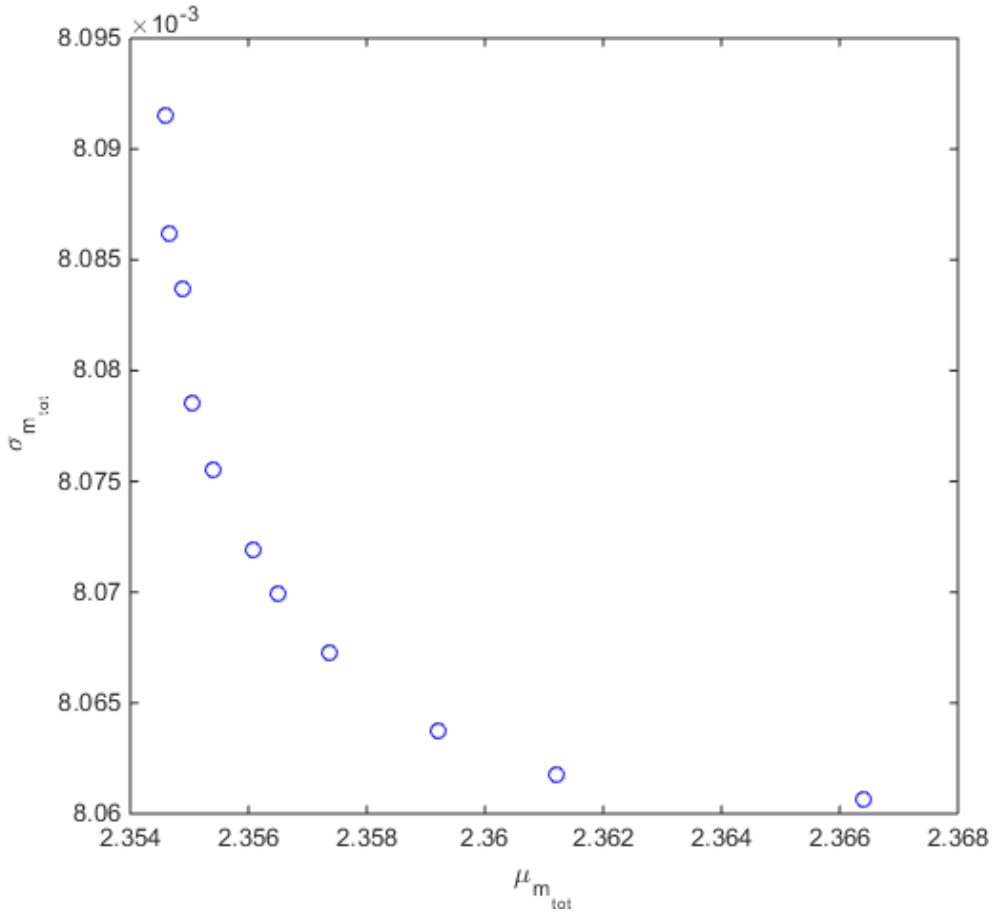


Fig. 3.13. Results of transformer with different value of weight using NTB

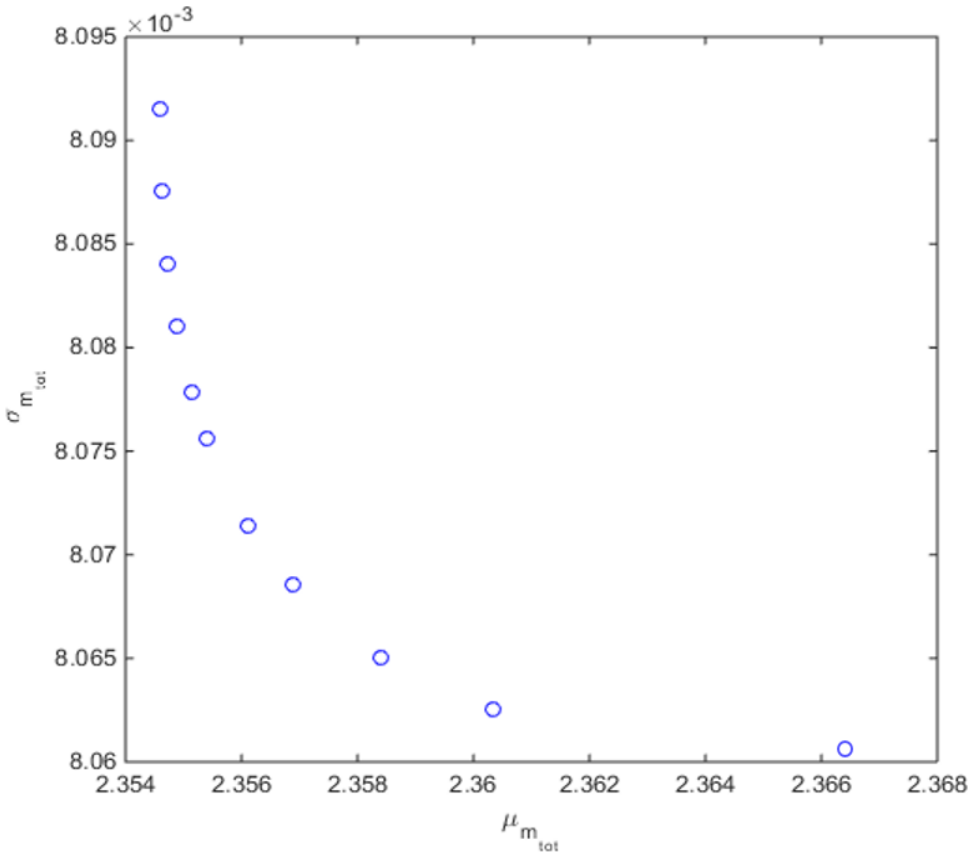


Fig. 3.14. Results of transformer with different value of weight using STB

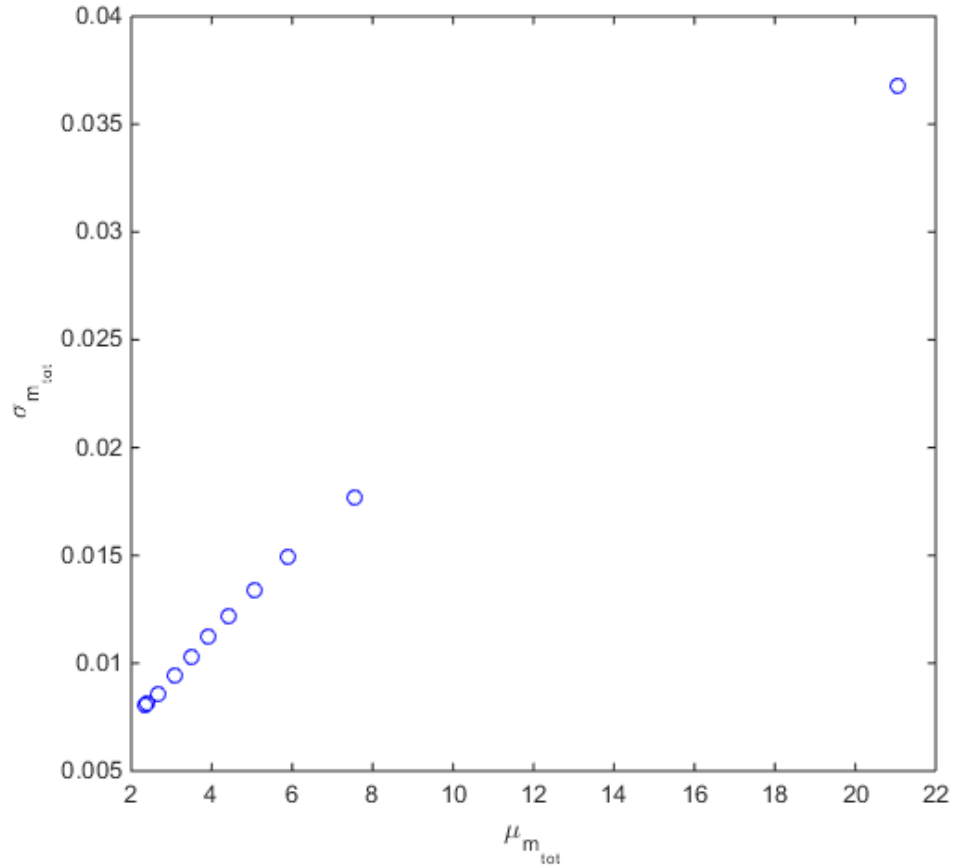


Fig. 3.15. Results of transformer with different value of weight using LTB

As can been seen from Fig 3.13 and Fig 3.14, the Pareto front of $\mu_{m_{tot}}$ and $\sigma_{m_{tot}}$ of transformer is convex, so not like in the simple mathematic example, this time STB it can also obtain a Pareto front as NTB.

And for LTB, it finds a front quite strange, in fact it is due to its objective. It always aims to maximize the mean while minimizing the standard deviation, thus it is not suitable for the minimization problems.

In conclusion, NTB and STB are most suitable, and in Fig. 3.16 we put the results of NTB and STB together, star marks present the solutions of NTB while circles present STB's. It can be seen that all the solutions are on the same Pareto front and it is quite difficult to say which one performs better.

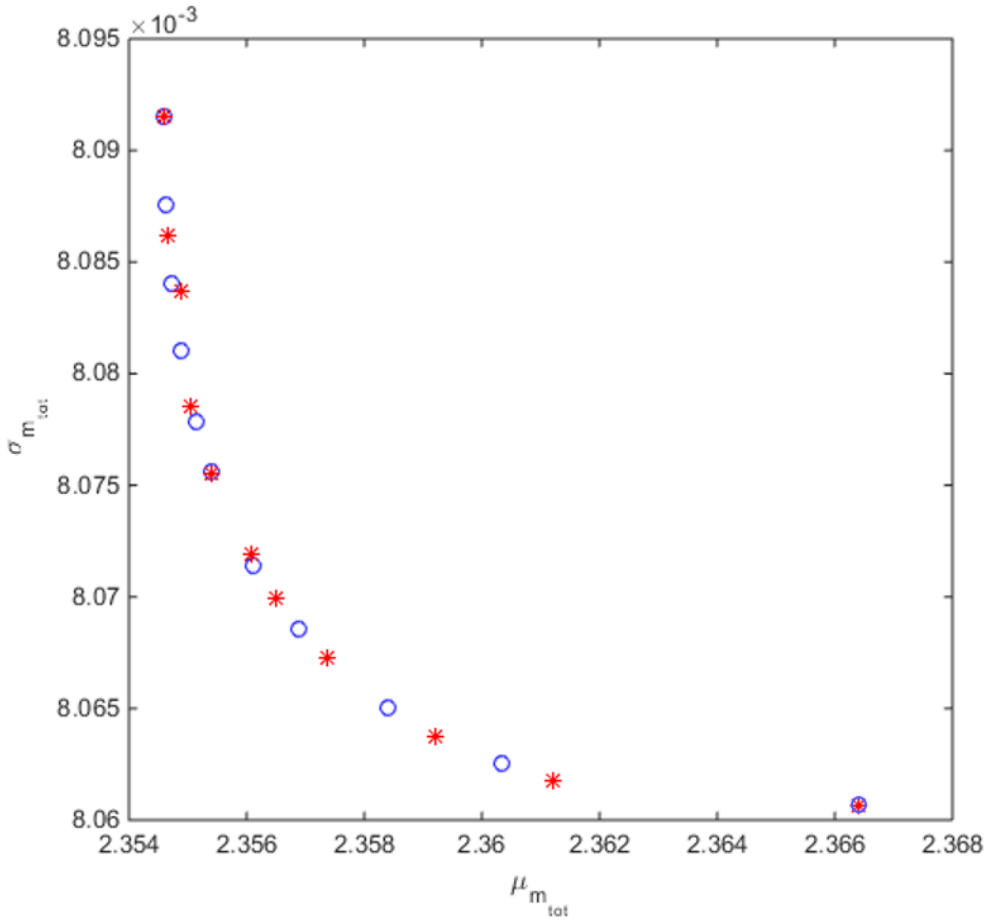


Fig. 3.16. Comparison the results of NTB and STB

Moreover, from the figure we can noticed that the solutions of STB are the same as those of NTB at the bounds $\omega_1 = 0, \omega_2 = 1$ or $\omega_1 = 1, \omega_2 = 0$. At the two bounds, the two formulations have same solutions. It is because in the condition $\omega_1 = 0, \omega_2 = 1$, both of their robust objective become $\left(\sigma_f / \sigma_{f_0}\right)^2$ according to the equations (1.61) and (1.62), since σ_{f_0} is fixed, it always leads to the same results. And for the other bound, as the target

f_t cannot be achieved, both of them try to minimize the $\mu_{m_{tot}}$ only and therefore the solutions are same.

Therefore, the conclusions of RBRDO methods are: Among the five formulations, NTB and STB are the most suitable for minimization problems, but the choice also needs to be based on the specific situation. Generally NTB is more universal as it can get a Pareto front in most situations while STB depends on the Pareto front's curvature. However, the problem of NTB is that we need a target value which may difficult to choose at first.

4 Comparison of methods for heavy models

In this part, each method for heavy models will be tested with FEM. As this model is complicated and highly constrained, the feasible regions are small and maybe discontinuous. After testing, more than a thousand of initial samples are needed, if the initial samples are not enough, there may be no point lying in the feasible region. Moreover, there are 7 input variables, small group of initial samples could not cover many domains thus the meta-model will be too inaccurate to use.

With these reasons, 7000 design points are chosen as initial samples to construct initial meta-models. These 7000 points are evaluated in parallel on 24 cores in about 49 hours. Then the following infill sampling points are evaluated sequentially.

The following tables show optimal values, objective, constraints or probabilities of failure, and the number of evaluations for each type of method. Results of method for heavy models with AM are also presented for comparison. However, when the same solution is reevaluated with FEM (the right column for each table), it can be seen that not all constraints are satisfied. So the methods cannot be performed with AM only. The mass computed with FEM is slightly different from the one with AM and the reason has been explained in section 1.4 in this Chapter.

4.1 WCO methods

For WCO, as presented in Chapter 2, WWCO is the most suitable method which can balance the speed of converge and the precision. So the principle of WWCO is used here for the heavy models. Moreover, after the test with mathematic example, it is concluded that the strategy with WCEI as ISC presented in Equation (2.53) could achieve a solution more accurate, and this ISC acts on meta-models of original objective and constraints to predict the worst-case. The infill points are found and added into the sample set at each iteration after the optimization:

$$\begin{aligned} &\max WCEI(\boldsymbol{d}) \\ &s.t.\ \ \widehat{\boldsymbol{g}}_{\boldsymbol{w}} \leq 0 \end{aligned} \tag{3.18}$$

where $\widehat{\boldsymbol{g}}_{\boldsymbol{w}}$ is calculated by $\max_{\boldsymbol{\xi} \in \boldsymbol{U}(\boldsymbol{d})} \hat{g}(\boldsymbol{d}+\boldsymbol{\xi})$, $\boldsymbol{d} = [a, b, c, d, n_1, S_1, S_2]^T$, $\boldsymbol{U}(\boldsymbol{d})$ is the uncertainty set around $\boldsymbol{d}$.

The results of meta-models based WWCO with criterion WCEI are shown in Table 3.8, for comparison, solutions of both FEM and AM are presented. The evaluation numbers are 7616 for FEM and 7466 for AM. These are amounts of 7000 initial samples and several hundreds of infill samples added by ISC. It seems that with AM it needs less evaluations, but in table 3.8, it could be seen that the first two constraints are violated if the results of AM are used directly with FEM. These two constraints refer to the copper and iron temperatures, as has been discussed, AM simplifies the temperature and treated them as uniform, it results in the underestimate of constraints. With the same reason, the objective value found is higher with FEM than AM.

Table 3.8. WCO results of transformer optimization with meta-model

Values	WWCO + FEM + meta-model	WWCO + AM + meta-model	FEM reevaluation
$a(mm)$	13.417	12.828	
$b(mm)$	54.096	54.331	
$c(mm)$	16.199	16.671	
$d(mm)$	42.794	43.048	
n_1	646.97	665.19	
$s_1(mm^2)$	0.3171	0.3267	
$s_2(mm^2)$	2.9509	2.9568	
$\mu_{m_{tot}}(kg)$	2.4445	2.3956	2.3992
$\sigma_{m_{tot}}(kg)$	0.0090	0.0082	0.0086
f_w	2.5041	2.4510	2.4573
g_{w_1}	-0.0021	-0.0167	0.1349
g_{w_2}	0	0	0.2139
g_{w_3}	-0.0714	-0.0289	-0.0776
g_{w_4}	0	0	-0.0396
g_{w_5}	-0.0055	-0.0036	0.0349
g_{w_6}	0	0	0.0228
g_{w_7}	-0.0864	-0.0621	-0.0902
Evaluations	7000+616	7000+466	1

4.2 RDO methods

For RDO, the formulation used is with the scaled values and weight parameters, the weight are all chosen as 0.5. Also following the conclusion in Chapter 2, the double-layer kriging based RDO will be used here. First layer is the meta-models of original objective $\hat{f}$ and constraints $\hat{\boldsymbol{g}}$, then they are used to compute the moments $\hat{\mu}_f, \hat{\boldsymbol{\mu}}_{\boldsymbol{g}}, \hat{\sigma}_f$ and $\hat{\boldsymbol{\sigma}}_{\boldsymbol{g}}$ by the uncertainty propagation methods like first order Taylor based method. At last criterion MWEI will be used to add new samples:

$$\max MWEI_{f_r}(\boldsymbol{d})$$
$$s.t.\ \hat{\boldsymbol{\mu}}_{\boldsymbol{g}}(\boldsymbol{d}) + k\,\hat{\boldsymbol{\sigma}}_{\boldsymbol{g}}(\boldsymbol{d}) \leq 0 \tag{3.19}$$

where the confidence level is $k = 3$ in this example.

In fact as the most effective method concluded before, ϵ-constraint can also be used here. The only difference is that in each iteration, at the beginning of searching infill points, it is necessary to use the meta-model to find the values of σ_{fmin} and σ_{fmax}, then add them into the constraints.

Results of FEM with the mentioned meta-model based RDO are presented in Table 3.9, 7273 total evaluations are needed for FEM. Among them, 7000 are initial samples which are chosen at the beginning, and 273 are added by MWEI. Same method with AM is also tested here and there are less iterations used with AM than with FEM, only 82 infill points are added during the optimization with AM. However, same with WCO, when the results of AM are reevaluated by FEM directly, it can be seen that the first two constraints are violated and objective value is augmented. Thus, FEM increases the precision of constraints and objective but this may also increase the evaluation number to find the optimum.

Table 3.9. RDO results of transformer optimization with meta-model

Values	Formulation 7 + FEM + meta-model	Formulation 7 + AM + meta-model	FEM reevaluation
$a(mm)$	13.037	13.033	
$b(mm)$	50.591	50.448	
$c(mm)$	16.742	16.877	
$d(mm)$	43.641	43.690	
n_1	646.87	638.62	
$s_1(mm^2)$	0.3216	0.3247	
$s_2(mm^2)$	2.9163	2.9140	
$\mu_{m_{tot}}(kg)$	2.3665	2.3624	2.3656
$\sigma_{m_{tot}}(kg)$	0.0085	0.0081	0.0081
g_{r_1}	-0.0074	-0.0087	0.0234
g_{r_2}	0	0	0.0208
g_{r_3}	-0.2263	-0.1739	-0.0823
g_{r_4}	0	0	0.0045
g_{r_5}	0	0	-0.0002
g_{r_6}	0	0	0.0089
g_{r_7}	-0.0918	-0.0906	-0.0766
Evaluations	7000+273	7000+82	1

4.3 RBDO methods

Among all the mentioned RBDO methods, SORA is the most universal and as has been proved in Chapter 2, the strategy SORA3 is the most efficient one without losing precision. Thus, SORA3 is chosen here to treat the transformer problem.

Also 7000 initial samples are used to create the meta-models of original objective and constraints, then infill points are selected by maximizing MWEI to find the deterministic optimum:

$$\begin{aligned} &\max_{\boldsymbol{d}} MWEI_f(\boldsymbol{d}) \\ &s.t.\ \hat{\boldsymbol{g}}(\boldsymbol{d}) \leq 0 \end{aligned} \tag{3.20}$$

After the deterministic optimum is found, sequential reliability assessment begins. Two different approach to analyze the reliability are used, if the inequality $\left\|\left(\boldsymbol{d}^k - \boldsymbol{d}^i\right)/\boldsymbol{\sigma}\right\| < \beta_t$ holds, where $\boldsymbol{d}^i$ is the optimum found by i-th cycle of deterministic optimization $(i = 1, \dots, k)$, then we consider that the meta-model around $\boldsymbol{d}^k$ is quite accurate as there already exists other samples. In this condition, reliability assessment will be done with meta-models directly:

$$\begin{aligned} \boldsymbol{x^k} &= \operatorname{argmax}_{\boldsymbol{x}} \hat{\boldsymbol{g}}(\boldsymbol{d}) \\ s.t.\ &\left\|\left(\boldsymbol{x} - \boldsymbol{d^k}\right)/\boldsymbol{\sigma}\right\| = \beta_t \end{aligned} \tag{3.21}$$

where $\boldsymbol{x}\sim N(d, \sigma^2)$. Then the results $\boldsymbol{x^k}$, also known as MPTPs are added to the sample set in order to evaluate and rebuild the meta-model $\hat{\boldsymbol{g}}$.

If the former inequality $\left\|\left(\boldsymbol{d}^k - \boldsymbol{d}^i\right)/\boldsymbol{\sigma}\right\| < \beta_t$ does not hold, then EI is used to add more samples in this region around $\boldsymbol{d}^k$ so as to get more accurate MPTPs:

$$\begin{aligned} \boldsymbol{x^k} &= \operatorname{argmax}_{\boldsymbol{x}} EI_g(\boldsymbol{d}) \\ s.t.\ &\left\|\left(\boldsymbol{x} - \boldsymbol{d^k}\right)/\boldsymbol{\sigma}\right\| = \beta_t \end{aligned} \tag{3.22}$$

With the MPTPs, the shift vector is calculated $\boldsymbol{t^{k+1}} = \boldsymbol{d^k} - \boldsymbol{x^k}$, then deterministic optimization considering this shift is introduced in but this time there is no more ISC:

$$\begin{aligned} \boldsymbol{d}^{k+1} &= \operatorname{argmin}_{\mathrm{d}} \hat{f}(\boldsymbol{d}) \\ s.t.\ &\hat{\boldsymbol{g}}\left(\boldsymbol{d} - \boldsymbol{t^{k+1}}\right) \leq 0 \end{aligned} \tag{3.23}$$

If the problem converges then $\boldsymbol{d}^{k+1}$ is considered as the optimum, if not, $(k+1)$-th cycle with reliability assessment will be continued.

In this transformer problem with FEM, 265 additional points are infilled during deterministic optimizations and reliability assessments. Among them, 42 points are added in the first deterministic optimization with MWEI, and all the rest are added by reliability assessments and deterministic optimization with meta-models in 8 cycles.

The result of SORA3 with FEM is shown in the following table, it leads to probabilities of failure close to their target value. The probabilities of failure are calculated by MCS with 10^6 samples, as the computing time is unbearable if all those 10^6 samples are evaluated with FEM, they are computed with meta-model.

Result of SORA3 with AM is also presented here, the objective value is always higher with FEM because AM underestimates constraints. With the same solution reevaluated by FEM, the highest probability of failure is 90% which violate greatly our target.

Table 3.10. RBDO results of transformer optimization with meta-model

Values	SORA3 + FEM	SORA3 + AM	FEM reevaluation
$a(mm)$	12.902	13.153	
$b(mm)$	46.042	51.039	
$c(mm)$	18.183	16.532	
$d(mm)$	42.318	43.098	
n_1	659.06	641.75	
$s_1(mm^2)$	0.3254	0.3216	
$s_2(mm^2)$	2.7552	2.8956	
$\mu_{m_{tot}}(kg)$	2.4028	2.3552	2.3520
$\sigma_{m_{tot}}(kg)$	0.0088	0.0081	0.0081
$P(T_{co} > 120℃)(\%)$	0%	0%	0%
$P(T_{ir} > 100℃)(\%)$	0.1506%	0.1567%	90.05%
$P\left(I_{\mu}/I_1 > 0.1\right)(\%)$	0%	0%	0%
$P(\Delta V_2/V_2 > 0.1)(\%)$	0.1348%	0.1420%	0.3281%
$P(f_1 > 0.5)(\%)$	0.1327%	0.1236%	71.20%
$P(f_2 > 0.5)(\%)$	0.1282%	0.1307%	0.0014%
$P(\eta < 0.8)(\%)$	0%	0%	0%
Evaluations	7000+265	7000+242	1

4.4 RBRDO methods

For RBRDO, in previous section we noticed that both NTB and STB work well with AM, and each of them have their own merits. Here for FEM, as the target value f_t is unknown, STB is chosen to use. The weights ω_1, ω_2 are all fixed as 0.5 and the scaled values are calculated by one of the initial samples.

The strategy of meta-model based STB is a combination of RDO's and RBDO's strategies. So as to choose the most suitable method, double-layer meta-model and SORA3 are brought in. The process is almost the same with SORA3, the only difference is that we use $\hat{f}_r$ instead of $\hat{f}$ in Equations (3.20) and (3.23), and before these two optimizations, an

additional step to build the second-layer meta-model, thus the meta-models of moments, should be added.

Table 3.11 shows the results of STB with FEM and AM using meta-models, respectively. And as always, the constraints are not satisfied when solution of AM is reevaluated with FEM. Moreover, for STB with meta-model of FEM, it uses 7541 points to converge, among them, 7000 are initial points, 134 are added by the first deterministic optimization, and others are infilled in 12 sequential circles of reliability assessment and optimization. Comparing with RBDO, it needs more samples to obtain the first deterministic optimum and more iterations to converge.

Table 3.11. RBRDO results of transformer optimization with meta-model

Values	STB + FEM + meta-model	STB + AM + meta-model	FEM reevaluation
$a(mm)$	13.042	12.933	
$b(mm)$	49.355	46.950	
$c(mm)$	18.104	19.267	
$d(mm)$	43.540	42.858	
n_1	648.52	652.99	
$s_1(mm^2)$	0.3256	0.3393	
$s_2(mm^2)$	2.9166	3.0422	
$\mu_{m_{tot}}(kg)$	2.4113	2.3662	2.3700
$\sigma_{m_{tot}}(kg)$	0.0088	0.0081	0.0081
$P(T_{co} > 120℃)(\%)$	0	0	76.87%
$P(T_{ir} > 100℃)(\%)$	0.1419%	0.1321%	53.14%
$P\left(I_{\mu}/I_1 > 0.1\right)(\%)$	0	0	0
$P(\Delta V_2/V_2 > 0.1)(\%)$	0.1395%	0.1371%	0.1707%
$P(f_1 > 0.5)(\%)$	0.1339%	0.1417%	0.2033%
$P(f_2 > 0.5)(\%)$	0.1341%	0.1337%	34.22%
$P(\eta < 0.8)(\%)$	0	0	0
Evaluations	7000+541	7000+493	1

From all above results, it can be seen that the first advantage of methods for heavy models with FEM is that a significant computing time can be saved as it reduces the number of evaluations. If optimization methods are used on FEM directly without meta-model, it can be imaged that more than 10^4 evaluations are needed by considering the multi-starts. But with meta-model, this number can be controlled in a few thousands. Of course multi-starts are also needed for models with meta-model when ISC are used to find the next sample point, but meta-model insures that no more evaluations are made in this process. Moreover, methods for heavy models could evaluate the initial samples in parallel when building the initial meta-model while all evaluations are calculated in sequential by methods for fast models.

The second and also the most important advantage is that kriging model gives accurate derivatives that enable the use of fast gradient-based algorithm. Contrarily, as FEM

provides noisy derivatives it requires a noise-free costly algorithm when directly connected with it.

5 Conclusion

In this chapter, a single-phase safety isolating transformer is introduced and a design optimization problem with 7 geometric variables is used as an example to test all the mentioned approaches for fast and heavy models using analytic model and finite element model, respectively. AM operates much faster than FEM but it is more imprecise than the latter as it neglects some thermal and magnetic information. Thus will change the domain of security and obtain some improper solutions. In that case, FEM is necessary but owing to the huge computational burden, only meta-model based methods are adapted.

First, according to the AM of transformer, the methods for fast models are verified and a more general conclusion can be made.

For WCO, the traditional WCO is the most time consuming approach, GWCO is the fastest but cannot lead to an accurate solution due to its principle. Among them, WWCO is the most suitable.

For RDO, multi-objective methods can find different solutions which satisfy different compromises between the mean and the standard deviation of objective. However, different algorithms for multi-objective have distinct applications: ϵ-constraint can apply to most occasions, but it needs additional information such as the minimum or maximum of the second objective. Formulation with scaled values and weight parameter depends on the Pareto front curvature which is not available in advance: if it is convex, it can obtain a complete Pareto front, if not then only the bounds may be found. Multi-objective algorithms like NSGA2 require large number of evaluations that limits their use.

For RBDO, single-loop methods are the fastest but not accurate enough. Double-loop methods needs much more evaluations. And for sequential decoupled methods, SAP approach is the most efficient one but sometimes it may fail to converge. Therefore, SORA is the most practical RBDO method.

For RBRDO, with regard to minimizing problem, both Normal-the-best and the smaller-the-better can be adapted, but for the former one, it needs to choose a target value before the process of optimization and for the latter one, it can obtain a complete Pareto front only if the front is convex. Considering that the shape of front is unknown in advance, Normal-the-better is more universal.

Then FEM is tested with the methods for heavy models. Results show their applicability in dealing with this highly constrained problem, and a significant computing time can be saved. Compared with AM of the same device, these approaches with FEM could get more accurate solutions.

Meta-model based WCO with WCEI is tested with FEM, and the optimum found is larger than using AM, thus because the inaccuracy of AM underestimate the constraints and enlarge the security domain. This phenomenon also exists for other categories of methods.

Then for RDO, as presented in Chapter 2, double-layer meta-model based method using MWEI is more effective so that it is chosen to be used here, and it should be noticed that as this method needs two layers of meta-model to build robust objective and constraints, the inaccuracy will also increase.

Meta-model based SORA is chosen to be used here for FEM, the strategy comprises two ISC and a proximity criterion. First MWEI is used during the primary deterministic optimization and then a proximity criterion before each reliability assessment, if it is satisfied, EI will be used during reliability assessment to add more samples around the current design point in order to find a more accurate probability.

At last for RBRDO, double-layer kriging method presented in RDO is combined with SORA in RDO. Results show that it could achieve the optimum in a dozen of iterations.

However, as the number of added samples are less than one tenth of the total number of evaluations, the choice of initial samples may be very unreasonable. The number can be reduced probably and a more suitable method to choose the positions of initial samples should be chosen. And as this model is complex, it can be imagined that it is very dependent on the initial samples. With different initial samples, the surrogate model may be different and it may leads to a disparate results if the meta-model is not accurate enough.

Conclusion and perspectives

The research work presented in this manuscript addresses the methods for the design optimizations with uncertainty. In real world applications, the uncertainties can be brought in by the model structural, observation, inaccuracy of parameters like measurement of geometric dimensioning and other factors we could not control. What we focus in this research is the uncertainty of geometric parameters. This uncertainty can have great effects on the product performance, cost and durability especially for sophisticated devices. In order to reduce these unexpected impacts, stochastic design optimizations (SDO) are proposed to treat the random variables and to get a solution more reliable and robust.

The presentation of the research work starts with the introduction of different SDO methods for fast models. The SDO can be mainly divided into four categories, worst-case optimization (WCO), robust design optimization (RDO), reliability-based design optimization (RBDO) and the combination of the latter two, reliability-based robust design optimization (RBRDO). The principles and emphases of these categories are different but their purposes are same: they try to find solutions shifted from the original deterministic solution lying on limit-states to the deep of security domain. That means their solutions have less probability to violate the constraints. Each of these categories has various approaches or formulations.

A simple mathematic model and a safety transformer such as an optimization problem in practical application are used in this manuscript to test the different approaches, first the mathematic example and the analytic model of transformer are tested with the methods for fast models, a synthesis is made to compare the results for each type of SDO and the most efficient method is chosen.

For WCO, it creates an uncertainty set around the current values of inputs, and the worst (maximum) values of objective function and constraints calculated in this set are treated as the worst-case outputs. So that the aim of WCO is to make sure even the worst-case outputs can respect the constraints, in this way the robustness can be improved. Traditional WCO, worst-vertex WCO and gradient-based approach are introduced and with the help of the simple mathematic model, it can be seen that worst-vertex WCO converges faster than traditional WCO and gets a solution more precise than gradient-based one, it is considered as the most efficient approach in this category.

RDO focuses on finding a feasible solution with minimum variability of original objective function. To tackle this problem, the mean and the standard deviation of original objective are both taken into account. The RDO approaches presented in this manuscript can be roughly divided into two types. The first one is mono-objective like Formulation 1 to 3 in Chapter 1, it fixes the relation between the mean and standard deviation and could

find only one solution. The rest formulations belong to multi-objective approach, they treat the mean and standard deviation as two different objectives so that we could get a Pareto front with them. Moreover, for solving the multi-objective problem, different techniques are proposed. Formulation 4 is ϵ-constraint method which changes the multi-objective problem into a series of mono-objective ones. Formulation 5 to 7 are weighted calculation which combine the two objectives together by using different weights. Formulation 8 and 9 are not transformed and require multi-objective algorithms like NSGA2 directly. Among all multi-objective formulations, ϵ-constraint method performs best as it can always get a Pareto front. Weighted formulations depend on the curvature of front, if it is convex the formulations can obtain a complete Pareto front, if not then only the bounds are found. And multi-objectives needs much more evaluations than others.

RBDO methods use probability of failure to describe the possibility of violating constraints. To calculate the probability, double-loop methods use a nested optimization to analyze the reliability, single-loop methods decide to use approximations of the reliability analysis, and sequential methods change the bi-level optimization into a series of decoupled deterministic optimization and reliability assessment. With the results of the two examples, it could conclude that for double-loop methods, PMA has higher rate of converge and less number of evaluations than RIA, but in general, they are time consuming. For single-loop methods, AMA is faster but much less accurate, SLA presents a good compromise between the accuracy and the speed. Sequential methods are faster than double-loop methods and does not lose any accuracy. Among them, SORA is the most effective one which could be applied in most cases.

The objectives of RBRDO consider also the mean and standard deviation of original objective function like RDO while the constraints turns into probabilistic ones like RBDO. As it combines both features of the other two categories, it improves the reliability and the robustness at the same time. Also, multi-objective and mono-objective approaches are proposed but a Pareto front obtained by multi-objective is obviously more universal. Among them, the Normal-the-best and Smaller-the-better formulations are both useful with their own merits.

Then new adaptive kriging based SDO methods with infill searching criteria (ISC) are proposed for time-consuming models like finite element analysis of electromagnetic devices. For those heavy models, the mentioned approaches in Chapter 1 are no longer suitable as the computational burden is unbearable. Moreover, all the mentioned methods need to calculate the gradient but, as finite element models provide noisy derivatives, it requires a noise-free costly algorithm when directly connected with it. So that kriging meta-models is a good alternative as it gives accurate derivatives that enable the use of fast gradient-based algorithm.

Although kriging methods have been applied in deterministic optimization for more than twenty years, it has not been applied very much into stochastic optimization in electromagnetic domain, especially with other three categories expect RBDO. Traditional

kriging approaches build a surrogate model at the beginning and then use it in optimization process without any change. These approaches may result in two outcomes: if the samples selected to build the meta-model are too many, it wastes the time, but if the samples are not enough, we may miss the good solution as the meta-model is not accurate. Thus, ISC are used to replace the objective function in optimization process. With the help of ISC, we can add the samples at the most needed place and optimize the problem while improving the accuracy simultaneously. After the new samples are added, meta-model is reconstructed and optimization continues until an accurate global solution is found.

Strategies of adaptive meta-model based on different SDO methods are proposed and with the help of both mathematic model and finite element model of the transformer, we highlight which ISC should be used and where the samples should be added during the process.

For WCO, meta-models are built on the original objective function and constraints in order to improve the accuracy of searching the worst-case. Then two criteria are proposed to act directly on the meta-model to estimate the worst-case values. After comparing their solutions and numbers of evaluations, it can be seen that the one enrich with WCEI is more accurate.

Then for RDO, as the calculations of moments of original objective and constraints need large amount of evaluations, a double-layer kriging method is proposed in order to save time. The first layer is the surrogate models for original objective and constraints and the second layer meta-models are for the moments which could be built with the former layer. Formulations and ISC like MWEI can be used with this estimated moments to solve the problem. Results of the two examples show that the number of evaluation is reduced greatly by this strategy than single-layer meta-model method without losing much accuracy.

Different approaches using SLA, PMA and SORA separately are proposed for RBDO. SLA uses MWEI to add new samples directly with meta-models of objective and constraints, but it is not accurate at all. Strategies based on PMA are either too time consuming or too inaccurate. After tests it could be found that the strategy based on SORA which uses MWEI for first deterministic optimization and a proximity criterion before reliability assessments is the most efficient one.

For RBRDO, as it has been seen in RDO, double-layer kriging method performs much better than single-layer kriging model, so it is also used here. This method is combined with process of SLA, PMA and SORA, and SORA performs better than other two.

Solutions of SDO with finite element model are obtained with the help of methods for heavy models. For these solutions, the objective value is generally higher than the one obtained by analytic model but this is due to the fact that the latter's inaccuracy underestimates some constraints.

In summary, this manuscript compares existing stochastic optimization methods for fast models and provides the most effective adaptive criterion-based kriging methods then verifies their feasibilities through an electromagnetic finite element model. The use of adaptive kriging models is inescapable for these expensive models, even though this thesis contributed to some adapted methods, a lot of works is still waiting in order to address a broader range of problems.

Firstly, in this research work, the proposed adaptive meta-model based SDO methods always add only one sample after each optimization with ISC. This may influence the speed of converge and the accuracy. How to infill more than one points with ISC in order to improve the speed and simplify the process is worth to be studied. For example, as the ISC is multimodal, after the global maximum is selected, we could focus on other regions far from this optimum in order to find several solutions. With this strategy, more than one points can be added and calculation burden does not increase too much.

Secondly, with the finite element model of transformer, 7000 initial samples are selected in this manuscript. However, these account for more than 90% of total devaluations, thus the initial samples and infilled samples are not balanced. There is a high probability that the number of initial samples can be smaller, so that a more suitable initial sample set should be chosen.

Thirdly, in this research, the uncertainties of inputs are assumed as independent and following the normal distribution. However, in real electromagnetic engineering field, these hypotheses may not hold, thus the methods should make the appropriate changes. For example, if the design variables follow the uniform distribution, then the Taylor-based method to calculate the moments of outputs in RDO is no longer applicable, another uncertainty propagation method need to be used. Also for calculating the probability of failure in RBDO, the form of iso-probalistic transformation should change in order to suit the first-order reliability method.

Reference

Alotto, P., Magele, C., Renhart, W., Weber, A., & Steiner, G. 2003. “Robust Target Functions in Electromagnetic Design.” COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 22 (3):549–60.

Amelin, M. 2004. “On monte carlo simulation and analysis of electricity markets.” PhD thesis, KTH.

Aoues, Y., & Chateauneuf, A. 2010. “Benchmark Study of Numerical Methods for Reliability-Based Design Optimization.” Structural and Multidisciplinary Optimization 41 (2):277–94.

Arsenyev, I., Duddeck, F., & Fischersworring-Bunk, A. 2015. “Surrogate-based Robust Shape Optimization for Vane Clusters.” 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA AVIATION Forum, (AIAA 2015-3361)

Asafuddoula, M., Singh, H. K., & Ray, T. 2015. “Six-Sigma Robust Design Optimization Using a Many-Objective Decomposition-Based Evolutionary Algorithm.” IEEE Transactions on Evolutionary Computation 19 (4):490–507.

Audet, C., Dennis, J., Moore, D., Booker, A., & Frank, P. 2000. “A Surrogate-Model-Based Method for Constrained Optimization.” *AIAA paper*, *4891*.

Baudoui, V. 2012. “Optimisation robuste multiobjectifs par modèles de substitution.” PhD thesis, Toulouse, ISAE.

Beyer, H. G., & Sendhoff, B. 2007. “Robust Optimization – A Comprehensive Survey.” Computer Methods in Applied Mechanics and Engineering 196 (33–34):3190–3218.

Bhamare, S. S., Yadav, O. P., & Rathore, A. 2009. “A Hybrid Quality Loss Function–based Multi-Objective Design Optimization Approach.” Quality Engineering 21 (3):277–289.

Bichon, B. J., Eldred, M. S., Swiler, L. P., Mahadevan, S., & McFarland, J. M. 2008. “Efficient Global Reliability Analysis for Nonlinear Implicit Performance Functions.” AIAA Journal 46 (10):2459–68.

Bompard, M. 2011. “ Modèles de substitution pour l'optimisation globale de forme en aérodynamique et méthode locale sans paramétrisation.” PhD thesis, Université Nice Sophia Antipolis.

Box, G. E., & Wilson, K. B. 1992. “On the experimental attainment of optimum conditions.” In *Breakthroughs in Statistics* (pp. 270-310). Springer New York.

Brochu, E., Cora, M., de Freitas, N. 2010. “A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement

Learning." Technical report, Department of Computer Science, University of British Columbia.

Broomhead, D. S., & Lowe, D. 1988. "Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks." Complex Systems, 2:321-323.

Chandra, M. J. 2001. Statistical Quality Control. CRC Press.

Chaudhuri, A., & Haftka, R. T. 2013. "A Stopping Criterion for Surrogate Based Optimization Using Ego." In 10th World Congress on Structural and Multidisciplinary Optimization, 20–24.

Chen, X., Hasselman, T. K., & Neill, D. J. 1997. "Reliability Based Structural Design Optimization for Practical Applications." In 38th Structures, Structural Dynamics, and Materials Conference. American Institute of Aeronautics and Astronautics.

Cheng, G., Xu, L., & Jiang, L. 2006. "A Sequential Approximate Programming Strategy for Reliability-Based Structural Optimization." Computers & Structures 84 (21):1353–67.

Cherkassky, V. 1997. "The Nature of Statistical Learning Theory." IEEE Transactions on Neural Networks 8 (6):1564.

Chiariello, A. G., Formisano, A., Martone, R., & Pizzo, F. 2015. "Gradient-Based Worst Case Search Algorithm for Robust Optimization." IEEE Transactions on Magnetics, 51(3), 1–4.

Choi, K. K., and Youn, B.D. 2001. "Hybrid Analysis Method for Reliability-Based Design Optimization." Journal of Mechanical Design, 125(2), 221-232.

Collette, Y., & Siarry, P. 2011. Optimisation multiobjectif: Algorithmes. Editions Eyrolles.

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.

David, L., Fyllingen, Ø., & Nilssona, L. 2008. "An Approach to Robust Optimization of Impact Problems Using Random Samples." Engineering Optimization 40 (11):989–1009.

Deb, K., & Gupta, H. 2006. "Introducing Robustness in Multi-Objective Optimization." Evolutionary Computation 14 (4):463–94.

Deb, K., Gupta, S., Daum, D., Branke, J., Mall, A. K., & Padmanabhan, D. 2009. "Reliability-Based Optimization Using Evolutionary Algorithms." IEEE Transactions on Evolutionary Computation 13 (5):1054–74.

Deb, K., & Gupta, H. 2005. "Searching for Robust Pareto-Optimal Solutions in Multi-Objective Optimization." In Evolutionary Multi-Criterion Optimization, edited by Carlos A. Coello Coello, Arturo Hernández Aguirre, and Eckart Zitzler, 3410:150–64. Berlin, Heidelberg: Springer Berlin Heidelberg.

Dersjö, T. 2012. “Methods for Reliability Based Design Optimization of Structural Components.” PhD thesis, KTH Royal Institute of Technology.

Di Barba, P. 2009. “Multiobjective Shape Design in Electricity and Magnetism. ” (Vol. 47). Springer Science & Business Media.

Doltsinis, I., Kang, Z., & Cheng, G. 2005. “Robust Design of Non-Linear Structures Using Optimization Methods.” Computer Methods in Applied Mechanics and Engineering 194 (12–16):1779–95.

Du, X., and Wei C. 1999. “Towards a Better Understanding of Modeling Feasibility Robustness in Engineering Design.” Journal of Mechanical Design 122 (4):385–94.

Du, X., and Wei C. 2004. “Sequential Optimization and Reliability Assessment Method for Efficient Probabilistic Design.” Journal of Mechanical Design 126 (2):225.

Du, X., Sudjianto, A., and Wei C. 2004. “An Integrated Framework for Optimization under Uncertainty Using Inverse Reliability Strategy.” Journal of Mechanical Design 126 (4):562–70.

Duff, I. S., & Reid, J. K. 1978. “An Implementation of Tarjan’s Algorithm for the Block Triangularization of a Matrix.” ACM Transactions on Mathematical Software (TOMS) 4 (2):137–147.

Duvenaud, D. K., Nickisch, H., & Rasmussen, C. E. 2011. “Additive Gaussian Processes.” In Advances in Neural Information Processing Systems, 226–234.

Enevoldsen, I. 1994. “Reliability-Based Optimization in Structural Engineering.” Structural Safety 15 (3):169–96.

Fonseca, C. M., Fleming, P. J., Zitzler, E., Deb, K., & Thiele, L. 2003. “Evolutionary Multi-Criterion Optimization.” Second International Conference, EMO 2003, Faro, Portugal, April 8-11, 2003: Proceedings. Lecture Notes in Computer Science 2632. Berlin ; New York: Springer.

Forrester, A. I., & Jones, D. R. 2008. “Global Optimization of Deceptive Functions with Sparse Sampling.” In 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. Vol. 1012. Victoria, Canada.

Friedman, J. H. 1991. “Multivariate Adaptive Regression Splines.” The Annals of Statistics 19 (1):1–67.

Giunta, A. A. 1997. “Aircraft Multidisciplinary Design Optimization Using Design of Experiments Theory and Response Surface Modeling Methods.” PhD thesis, Virginia Tech, 1997.

Glancy, C. G. 1999. “A Second-Order Method for Assembly Tolerance Analysis.” In In Proceedings of the ASME Design Automation Conference. Las Vegas, Nevada.

Grandhi, R. V., & Wang, L. 1998. “Reliability-Based Structural Optimization Using Improved Two-Point Adaptive Nonlinear Approximations.” Finite Elements in Analysis and Design 29 (1):35–48.

Hansen, C. B. 2007. “Generalized Least Squares Inference in Panel and Multilevel Models with Serial Correlation and Fixed Effects.” Journal of Econometrics 140 (2):670–94.

Hasofer, A. M., & Lind, N. C. 1974. “Exact and Invariant Second-Moment Code Format.” Journal of the Engineering Mechanics Division 100 (1):111–21.

Huang, D. 2005. “Experimental Planning and Sequential Kriging Optimization Using Variable Fidelity Data.” PhD thesis, The Ohio State University.

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.

Jin, R., Du, X., & Chen, W. “The Use of Metamodeling Techniques for Optimization under Uncertainty.” Structural and Multidisciplinary Optimization 25 (2):99–116.

Jones, D. R., Schonlau, M., & Welch, W. J. 1998. “Efficient Global Optimization of Expensive Black-Box Functions.” Journal of Global Optimization 13 (4):455–492.

Jurecka, F. 2007. “Robust Design Optimization Based on Metamodeling Techniques.” PhD thesis, Technische Universität München.

Kang, Z. 2005. “Robust Design Optimization of Structures under Uncertainties.” PhD thesis, Universität Stuttgart.

Kleijnen, J. P. 2009. “Kriging Metamodeling in Simulation: A Review.” European Journal of Operational Research 192 (3):707–16.

Kim, D. W., Choi, N. S., Choi, K. K., & Kim, D. H. 2015. “A Single-Loop Strategy for Efficient Reliability-Based Electromagnetic Design Optimization.” IEEE Transactions on Magnetics 51 (3):1–4.

Krige, D. G. 1951. “A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand.” Journal of the Southern African Institute of Mining and Metallurgy 52 (6):119–139.

Kruskal, W. 1968. “When Are Gauss-Markov and Least Squares Estimators Identical? A Coordinate-Free Approach.” The Annals of Mathematical Statistics 39 (1):70–75.

Kuschel, N., & Rackwitz, R. 1997. “Two Basic Problems in Reliability-Based Structural Optimization.” Mathematical Methods of Operations Research 46 (3):309–33.

Lai, X.M., Lai, Q.F., Huang, H., Wang, C., Zhang, Y., Yan, L., Yang, J.H., & Liao, S.R. 2016. “High-Efficient Sequential Approximate Strategy for Reliability-Based Robust Design

Optimization." Journal of Mechanical Engineering Research & Developments 39:278–295.

Lee, I., Choi, K. K., Du, L. 2008. "Dimension Reduction Method for Reliability-Based Robust Design Optimization." Computers & Structures 86 (13–14):1550–62.

Lee, K. H., & Kang, D. H. 2006. "A Robust Optimization Using the Statistics Based on Kriging Metamodel." Journal of Mechanical Science and Technology 20 (8):1169–82.

Lee, K. H., & Park, G. J. 2001. "Robust Optimization Considering Tolerances of Design Variables." Computers & Structures 79 (1):77–86.

Lee, K. H., & Park, G. J. 2006. "A Global Robust Optimization Using Kriging Based Approximation Model." JSME International Journal Series C 49 (3):779–88.

Lei, J., Lima-Filho, P., Styblinski, M. A., & Singh, C. 1998. "Propagation of Variance Using a New Approximation in System Design of Integrated Circuits." In Proceedings of the IEEE 1998 National Aerospace and Electronics Conference. NAECON 1998. Celebrating 50 Years (Cat. No.98CH36185), 242–46.

Lettvin, J. Y., Maturana, H. R., McCulloch, W. S., & Pitts, W. H. 1959. "What the Frog's Eye Tells the Frog's Brain." Proceedings of the IRE 47 (11):1940–51.

Li, W, and Li Y. 1994. "An Effective Optimization Procedure Based on Structural Reliability." Computers & Structures 52 (5):1061–67.

Li, Y., Rotaru, M., & Sykulski, J. K. 2016. "Kriging Based Robust Optimisation Algorithm for Minimax Problems in Electromagnetics." Archives of Electrical Engineering 65 (4):843–854.

Liang, J., Mourelatos, Z. P., & Nikolaidis, E. 2007. "A Single-Loop Approach for System Reliability-Based Design Optimization." Journal of Mechanical Design 129 (12):1215–24.

Liang, J., Mourelatos, Z. P., & Tu, J. 2004, January. "A single-loop method for reliability-based design optimization." In ASME 2004 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (pp. 419-430). American Society of Mechanical Engineers.

Lichtenstern, A. 2013. "Kriging Methods in Spatial Statistics." PhD thesis, Technische Universität München.

Liu, J., Han, Z. H., & Song, W. P. 2012. "Comparison of Infill Sampling Criteria in Kriging-Based Aerodynamic Optimization." In 28th Congress of the International Council of the Aeronautical Sciences, 23–28.

Liu, P. L., & Der Kiureghian, A. 1991. "Optimization Algorithms for Structural Reliability." Structural Safety 9 (3):161–177.

Liu, W., & Batill, S. M. (2000, September). “Gradient-enhanced neural network response surface approximations.” In Proceedings of the 8th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization (pp. 6-8).

Lizotte, D. J., Greiner, R., & Schuurmans, D. 2012. “An Experimental Methodology for Response Surface Optimization Methods.” Journal of Global Optimization 53 (4):699–736.

Locatelli, M. 1997. “Bayesian Algorithms for One-Dimensional Global Optimization.” Journal of Global Optimization 10 (1):57–76.

Lönn, David (né), A., Jergeus, J., & Nilsson, L. 2013. “Robust Optimization of Front Members in a Full Frontal Car Impact.” Engineering Optimization 45 (3):245–64.

Lopez, R. H., & Beck, A. T. 2012. “Reliability-Based Design Optimization Strategies Based on FORM: A Review.” Journal of the Brazilian Society of Mechanical Sciences and Engineering 34 (4):506–14.

Lophaven, Søren Nymand. 2002. “DACE - A Matlab Kriging Toolbox, Version 2.0.” URL: www2.imm.dtu.dk/projects/dace/dace.pdf

Luo, K., Wang, J., & Du, X. 2012. “Robust Mechanism Synthesis with Truncated Dimension Variables and Interval Clearance Variables.” Mechanism and Machine Theory 57:71–83.

Matheron, G. 1963. “Principle of Geostatistics.” Economic Geology 58:1246–66.

Mester, V. 2007. “Conception Optimale Systémique Des Composants Des Chaînes de Traction Electrique.” PhD thesis, Ecole Centrale de Lille.

Mohammadi, H. 2016. “Kriging-Based Black-Box Global Optimization: Analysis and New Algorithms.” PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne.

Moody, J., & Darken, C. J. 1989. “Fast Learning in Networks of Locally-Tuned Processing Units.” Neural Computation 1 (2):281–94.

Moustapha, M., Sudret, B., Bourinet, J. M., & Guillaume, B. 2016. “Quantile-Based Optimization under Uncertainties Using Adaptive Kriging Surrogate Models.” Structural and Multidisciplinary Optimization 54 (6):1403–1421.

Neittaanmäki, P., Rudnicki, M., & Savini, A. 1996. “Inverse Problems and Optimal Design in Electricity and Magnetism.” Oxford University Press.

Neubert, H., Kamusella, A., & Pham, T.-Q. 2010. “Robust and reliability-based design optimization of electromagnetic actuators using heterogeneous modeling with comsol multiphysics and dynamic network models.” In Excerpt from the Proceedings of the COMSOL Conference.

Nikolaidis, E., & Burdisso, R. 1988. “Reliability Based Optimization: A Safety Index Approach.” Computers & Structures 28 (6):781–88.

Orr, M. J. et al. 1996. Introduction to Radial Basis Function Networks. Technical Report, Center for Cognitive Science, University of Edinburgh.

Padulo, M., Forth, S. A., & Guenov, M. D. 2008. “Robust Aircraft Conceptual Design Using Automatic Differentiation in Matlab.” In Advances in Automatic Differentiation, 64:271–80. Berlin, Heidelberg: Springer Berlin Heidelberg.

Paiva, R. M. 2010. “A Robust and Reliability-Based Optimization Framework for Conceptual Aircraft Wing Design.” PhD thesis, University of Victoria.

Park, H. U., Lee, J. W., Chung, J., & Behdinan, K. 2015. “Uncertainty-based MDO for aircraft conceptual design.” Aircraft Engineering and Aerospace Technology, 87(4), 345–356.

Parkinson, A., Sorensen, C., & Pourhassan, N. 1993. “A General Approach for Robust Optimal Design.” Journal of Mechanical Design 115 (1):74–80.

Peter, J., Marcelet, M., Burguburu, S., & Pediroda, V. 2007. “Comparison of Surrogate Models for the Actual Global Optimization of a 2D Turbomachinery Flow.” In Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization, 46–51. World Scientific and Engineering Academy and Society (WSEAS).

Picheral, L. 2013. “Contribution to the Robust Preliminary Design in Product Engineering.” PhD thesis, Université de Grenoble.

Putko, M. M., Taylor, A. C., Newman, P. A., & Green, L. L. 2002. “Approach for Input Uncertainty Propagation and Robust Design in CFD Using Sensitivity Derivatives.” Journal of Fluids Engineering 124 (1):60.

Qu, X., & Haftka, R. T. 2004. “Reliability-Based Design Optimization Using Probabilistic Sufficiency Factor.” Structural and Multidisciplinary Optimization 27 (5).

Rackwitz, R., & Flessler, B. 1978. “Structural Reliability under Combined Random Load Sequences.” Computers & Structures 9 (5):489–494.

Rasmussen, C. E., & Williams, C. K. 2006. “Gaussian Processes for Machine Learning.” MIT Press.

Reddy, M. V., Grandhi, R. V., & Hopkins, D. A. 1994. “Reliability Based Structural Optimization: A Simplified Safety Index Approach.” Computers & Structures 53 (6):1407–1418.

Ren, Z., Pham, M. T., & Koh, C. S. 2013. “Robust Global Optimization of Electromagnetic Devices With Uncertain Design Parameters: Comparison of the Worst Case Optimization Methods and Multiobjective Optimization Approach Using Gradient Index.” IEEE Transactions on Magnetics 49 (2):851–59.

Ren, Z., M. T. Pham, M. Song, D. H. Kim, and C. S. Koh. 2011. “A Robust Global Optimization Algorithm of Electromagnetic Devices Utilizing Gradient Index and Multi-Objective Optimization Method.” IEEE Transactions on Magnetics 47 (5):1254–57.

Ren, Z., Zhang, D., & Koh, C. S. 2013. “New Reliability-Based Robust Design Optimization Algorithms for Electromagnetic Devices Utilizing Worst Case Scenario Approximation.” IEEE Transactions on Magnetics 49 (5):2137–40.

Rijpkema, J. J. M., Etman, L. F. P., & Schoofs, A. J. G. 2001. “Use of Design Sensitivity Information in Response Surface and Kriging Metamodels.” Optimization and Engineering 2 (4):469–484.

Roos, D., Adam, U., & Bucher, C. 2006. “Robust Design Optimization.” Proceedings. Weimarer Optimierung-Und Stochastiktage 3.

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. 1989. “Design and Analysis of Computer Experiments.” Statistical Science 4 (4):409–23.

Sasena, M. J., Papalambros, P. Y., & Goovaerts, P. 2000. “Metamodeling Sampling Criteria in a Global Optimization Framework.” 8thAIAA/NASA/USAF/ISSMO Multidisciplinary Analysis and Optimization Conference.

Sasena, M.J., Papalambros, P. Y., & Goovaerts, P. 2001. “The Use of Surrogate Modeling Algorithms to Exploit Disparities in Function Computation Time within Simulation-Based Optimization.” In In The Fourth World Congress of Structural and Multidisciplinary Optimization.

Sasena, M. J. 2002. “Flexibility and Efficiency Enhancements for Constrained Global Design Optimization with Kriging Approximations.” PhD thesis, University of Michigan.

Schonlau, M. 1997. “Computer Experiments and Global Optimization.” PhD thesis, University of Waterloo.

Séguier, G., & Notelet, F. 1994. “Electrotechnique Industrielle.” Paris: Editions TEC & DOC.

Shahraki, A. F., & Noorossana, R. 2013. “A combined algorithm for solving reliability-based robust design optimization problems.” Journal of Mathematics and Computer Science, 7, 54-62.

Shan, S., and Wang. G. G. 2008. “Reliable Design Space and Complete Single-Loop Reliability-Based Design Optimization.” Reliability Engineering & System Safety 93 (8):1218–30.

Shimoyama, K., Sato, K., Jeong, S., & Obayashi, S. 2012. “Comparison of the Criteria for Updating Kriging Response Surface Models in Multi-Objective Optimization.” In Evolutionary Computation (CEC), 2012 IEEE Congress on, 1–8. IEEE.

Simpson, T. W., Mauery, T. M., Korte, J. J., & Mistree, F. 2001. “Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization.” AIAA Journal 39 (12):2233–41.

Sóbester, A., Leary, S. J., & Keane, A. J. 2005. “On the Design of Optimization Strategies Based on Global Response Surface Approximation Models.” Journal of Global Optimization 33 (1):31–59.

Steiner, G., Weber, A., & Magele, C. 2004. “Managing Uncertainties in Electromagnetic Design Problems with Robust Optimization.” IEEE Transactions on Magnetics 40 (2):1094–99.

Sun, G., Li, G., Zhou, S., Li, H., Hou, S., & Li, Q. 2011. “Crashworthiness Design of Vehicle by Using Multiobjective Robust Optimization.” Structural and Multidisciplinary Optimization 44 (1):99–110.

Sundaresan, S., Ishii, K., & Houser, D. R. 1992. “Design optimization for robustness using performance simulation programs.” Engineering Optimization, 20(3), 163-178.

Sundaresan, S., Ishii, K., & Houser, D. R. 1995. “A Robust Optimization Procedure with Variation on Design Variable and Constraints.” Engineering Optimization 24 (2):101–17.

Todorovic, P.. 2012. “An Introduction to Stochastic Processes and Their Applications.” Springer Science & Business Media.

Tran, T. V., Brisset, S., & Brochet, P. 2007. “A Benchmark for Multi-Objective, Multi-Level and Combinatorial Optimizations of a Safety Isolating Transformer.” in COMPUMAG: 167-168.

Tran, T. V. 2009. “Problèmes Combinatoires et Modèles Multi-Niveaux Pour La Conception Optimale Des Machines Électriques.” PhD thesis, École Centrale de Lille.

Tu, J., Choi, K. K., & Park, Y. H. 1999. “A New Study on Reliability-Based Design Optimization.” Transactions-American Society of Mechanical Engineers Journal of Mechanical Design 121 (4):557–564.

Ur Rehman, S., & Langelaar, M. 2017. “Expected Improvement Based Infill Sampling for Global Robust Optimization of Constrained Problems.” Optimization and Engineering 18 (3):723–53.

Varadarajan, S., Chen, W., & Pelka, C. J. 2000. “Robust Concept Exploration of Propulsion Systems with Enhanced Model Approximation Capabilities.” Engineering Optimization A35, 32 (3):309–334.

Vazquez, E., Villemonteix, J., Sidorkiewicz, M., & Walter, E. 2008. “Global Optimization Based on Noisy Evaluations: An Empirical Study of Two Statistical Approaches.” In Journal of Physics: Conference Series, 135:012100. IOP Publishing.

Wang, Z., Huang, H. Z., & Liu, Y. 2010. “A Unified Framework for Integrated Optimization Under Uncertainty.” Journal of Mechanical Design 132 (5):051008.

Wu, Y. T., & Wang, W. 1998. “Efficient probabilistic design by converting reliability constraints to approximately equivalent deterministic constraints.” J. Integr. Des. Process Sci, 2(4), 13-21.

Wu, Y. T., Millwater, H. R., & Cruse, T. A. 1990. “Advanced Probabilistic Structural Analysis Method for Implicit Performance Functions.” AIAA Journal 28 (9):1663–69.

Xiao, S., Li, Y., Rotaru, M., & Sykulski, J. K. 2014. “Considerations of Uncertainty in Robust Optimisation of Electromagnetic Devices.” International Journal of Applied Electromagnetics and Mechanics 46 (2):427–36.

Xiao, S., Rotaru, M., & Sykulski, J. K 2012. “Exploration versus Exploitation Using Kriging Surrogate Modelling in Electromagnetic Design.” Edited by Slawomir Wiak. COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 31 (5):1541–51.

Xiao, S., Rotaru, M., & Sykulski, J. K. 2013. “Adaptive Weighted Expected Improvement With Rewards Approach in Kriging Assisted Electromagnetic Design.” IEEE Transactions on Magnetics 49 (5):2057–60.

Xinying L., Wang, S., Qiu, J., Zhu, J. G., Guo, Y., & Lin, Z. W. 2008. “Robust Optimization in HTS Cable Based on Design for Six Sigma.” IEEE Transactions on Magnetics 44 (6):978–81.

Yadav, O. P., Bhamare, S. S., & Rathore, A. 2010. “Reliability-Based Robust Design Optimization: A Multi-Objective Framework Using Hybrid Quality Loss Function.” Quality and Reliability Engineering International 26 (1):27–41.

Yi, P., Cheng, G., & Jiang, L. 2008. “A Sequential Approximate Programming Strategy for Performance-Measure-Based Probabilistic Structural Design Optimization.” Structural Safety 30 (2):91–109.

Youn, B. D., Choi, K. K., Yang, R. J., & Gu, L. 2004. “Reliability-Based Design Optimization for Crashworthiness of Vehicle Side Impact.” Structural and Multidisciplinary Optimization 26 (3–4):272–83.

Youn, B. D., & Choi, K. K. 2004a. “An Investigation of Nonlinearity of Reliability-Based Design Optimization Approaches.” Journal of Mechanical Design 126 (3):403.

Youn, B. D., & Choi, K. K. 2004b. “Selecting Probabilistic Approaches for Reliability-Based Design Optimization.” AIAA Journal 42 (1):124–131.

Youn, B. D., Choi, K. K., & Yi, K. 2015. “Performance Moment Integration (PMI) Method for Quality Assessment in Reliability-Based Robust Design Optimization.” Mechanics Based Design of Structures and Machines 33 (2):185–213.

Yu, H. 2011. “Reliability-Based Design Optimization of Structures : Methodologies and Applications to Vibration Control.” PhD thesis, Ecole centrale de Lyon.

Yu, X., Chang, K.H., and Choi, K. K. 1998. “Probabilistic Structural Durability Prediction.” AIAA Journal 36 (4):628–37.

Zhang, Y. 2015. “Reliability-based robust design optimization of vehicle components, Part II: Case studies.” Frontiers of Mechanical Engineering, 10(2), 145-153.

Zhou, A., and Zhang, Q. 2010. “A Surrogate-Assisted Evolutionary Algorithm for Minimax Optimization.” In IEEE Congress on Evolutionary Computation, 1–7.

Méthodes de conception par optimisation robuste et fiable de dispositifs électrotechniques

Résumé: Cette thèse porte sur les méthodes d'optimisation robuste et fiable. Les différentes catégories de méthodes d'optimisation stochastique pour traiter les incertitudes sur les dimensions et les matériaux sont présentées. Ces méthodes visent à trouver une solution plus robuste et/ou fiable en minimisant la variance de l'objectif et/ou en réduisant la probabilité de violer les contraintes de faisabilité. Cependant, ces méthodes augmentent le nombre d'évaluations par rapport à une optimisation déterministe et nécessitent le gradient qui peut être bruité pour des modèles éléments finis. Ainsi, des stratégies basées sur des méta-modèles de kriging sont proposées pour approcher des fonctions complexes et donner des dérivées sans bruit. Des critères d'enrichissement sont utilisés pour affiner la précision tout en cherchant l'optimum. Différentes stratégies comprenant le choix du critère et le positionnement de l'enrichissement sont comparées pour mettre en évidence les plus efficaces. Enfin, les stratégies d'optimisation développées dans cette thèse sont appliquées à l'optimisation d'un transformateur modélisé par des équations analytiques puis par la méthode des éléments finis.

Mots-clefs: optimisation, conception, incertitude, robustesse, fiabilité, méta-modèle, transformateur, modèle d'éléments finis

Methods for Robust and Reliability-based Design Optimization of Electromagnetic Devices

Abstract: This PhD thesis deals with the robust and reliability-based optimization problems under uncertainty on dimensions and material properties. First, the different categories of stochastic optimization methods to treat the uncertainty are presented. These methods aim to find a more robust and/or reliable solution by minimizing the variance of objective and/or reducing the probability to violate the constraints of feasibility. However, as these methods increase the number of evaluations compared to deterministic optimization and need the gradient information that may be noisy when provided by finite element models, they are not suitable for the time-consuming models. So kriging-based meta-model strategies are proposed as they can approximate complex functions and give noise-free derivatives. Infill sampling criteria are used to increase their precision while searching for the optimal solution. Different strategies including the choice of the criterion and the positioning of sample enrichment are compared to highlight the most effective ones. Then, the optimization approaches developed within this research work are applied to the optimization problem of a transformer modelled with analytic equations and finite element models.

Keywords: design optimization, uncertainty, robust, reliability, meta-model, electromagnetic device, finite element model