

Multipoint Approximation Method for Large Scale Design Optimization Problems Under Uncertainty

First I. Author*

*International Center for Numerical Methods in Engineering (CIMNE)
Universidad Politécnica de Cataluña, Campus Norte UPC, 08034 Barcelona, Spain
Email: cimne@cimne.upc.edu*

Second B. Author

*Spanish Association for Numerical Methods in Engineering (SEMNI)
Edificio C1, Campus Norte UPC, Gran Capitán s/n, 08034 Barcelona, Spain
Email: semni@cimne.upc.edu*

Summary

The purpose of this document is to provide an example of how authors must format their extended abstracts. The summary should be a very brief description of the work, which should enable an easy indexing. It should not exceed 200 words in length. It must be single spaced, even justified across the full width of the page.

Keywords: list up to 8 (eight) words or group of words in order of priority, separated by commas, which will be used for indexing purposes.

1 Introduction

In industrial design optimization problems, random inaccuracies of the production process may lead to the designer's inability to fully control the design variables. Therefore, there can be a discrepancy between the products *as designed* and *as produced* that results in actual performance being different from the performance expected at the design stage. Possible discrepancy between the design and the manufactured product becomes even more important if the latter violates some critical constraints, e.g. safety regulations.

In addition, the system's performance may depend on parameters of random nature that are beyond the designer's control, e.g. the operation conditions. Therefore, accounting for both of these types of uncertainties is crucial for achieving robust and reliable performance.

Optimisation under uncertainty deals with responses $F(\mathbf{x}, \mathbf{y})$ that depend on deterministic design variables \mathbf{x} and random 'environmental' variables \mathbf{y} . Uncertainty in the design variables can also be modelled by introducing additional random variables, e.g. as additive or multiplicative noise. In this case, the responses $F(\mathbf{x}, \mathbf{y})$ receive as input the sum $\mathbf{x} + \mathbf{u}$ (if the noise is additive), where \mathbf{u} is a random variable with a known probability distribution, yielding responses depending on \mathbf{x} , \mathbf{u} and \mathbf{y} . Randomness in \mathbf{u} and \mathbf{y} induces randomness in the responses $F(\mathbf{x} + \mathbf{u}, \mathbf{y})$. To convert them into deterministic

functions, that can be optimised, a mapping R , called the *risk measure*[2] or *robustness measure*[3], is applied to the responses. The result is then a deterministic function of the design variables: $\tilde{F}(\mathbf{x}) = R(F(\mathbf{x} + \mathbf{u}, \mathbf{y}))$. The choice of a particular risk measure R reflects the designer's attitude towards the risk of sub-optimal performance or constraint violation. For example, R can be a combination of the mean and standard deviation of $F(\mathbf{x} + \mathbf{u}, \mathbf{y})$ with respect to the joint probability distribution of \mathbf{u} and \mathbf{y} or a quantile of the distribution of $F(\mathbf{x} + \mathbf{u}, \mathbf{y})$, i.e. such a value that the random variable of $F(\mathbf{x} + \mathbf{u}, \mathbf{y})$ exceeds it with a given, typically small, probability.

Computations of the risk measures typically involve evaluation of high-dimensional integrals, which becomes a challenging task if the number of design variables and/or environmental variables is large. The problem becomes even more difficult if the original responses are computationally expensive, which is the case, for example, in computational fluid dynamics (CFD), where one function evaluation may take hours of even days. Therefore, direct computation of the risk measures using original responses becomes infeasible and therefore approximations of responses, also referred to as metamodels, need to be used.

In problems with a large (in the order of hundreds) number of design variables, the multipoint approximation method (MAM4,5,6) proved to be efficient, e.g. in turbomachinery applications7,8,9. This method is an

iterative optimization technique based on mid-range approximations built in trust regions. A trust region is a sub-domain of the design space in which a set of design points, produced according to a small-scale design of experiments (DoE), are evaluated. These and a subset of previously evaluated design points are used to build metamodels of the objective and constraint functions that are considered to be valid within a current trust region. The trust region will then translate and change size as optimization progresses. The trust region strategy has gone through several stages of development to account for the presence of numerical noise in the response function values^{10,11} and occasional simulation failures¹². The mid-range approximations used in the trust regions, as originally suggested in Ref. 4 for structural optimization problems, are intrinsically linear functions (i.e. nonlinear functions that can be led to a linear form by a simple transformation) for individual sub-structures, and assembly of them for the whole structure. This was enhanced by the use of gradient-assisted metamodels,⁶ use of simplified numerical models that is also termed a multi-fidelity approach,¹³ and the use of analytical models derived by Genetic Programming¹⁴. One of the recent developments¹⁵ involved the use of approximation assemblies, i.e. a two stage approximation building process that is conceptually similar to the original one used in Ref. 4 but is free from the limitation that lower level approximations are linked to individual substructures.

This paper presents a new development in the Multipoint Approximation Method that makes it capable of handling problems with uncertainty in design variables as well as in additional ‘environmental’ variables. The approach relies on approximations built in the combined space of design variables and environmental variables, and subsequent application of a risk measure and optimization with respect to the deterministic design variables, all within the iterative trust-region-based framework of MAM.

2 The Multipoint Approximation Method

It is useful to start with a brief description of the deterministic version of MAM. A typical formulation of a constrained optimization problem that MAM works with is as follows:

$$\begin{aligned} & \min_{a_i \leq x_i \leq b_i} F_0(\mathbf{x}) \\ & \text{s.t. } F_j(\mathbf{x}) \leq 1, \quad j = 1, \dots, M, \end{aligned} \quad (1)$$

where \mathbf{x} is the vector of design variables, \mathbf{a} and \mathbf{b} are the lower and upper bounds for the design variables, respectively, $F_0(\mathbf{x})$ is the objective function, and $F_j(\mathbf{x})$ are the constraints. The numbers of design variables and constraints are n and M , respectively. MAM attempts to solve this problem by using approximations of the objective function and constraints in a series of trust regions. The trust region strategy seeks to zoom in on the region where the constrained minimum is achieved. It aims at finding a trust region that is sufficiently small for the approximations

to be of sufficiently good quality to improve the design, and that contains the point of the constrained minimum as its interior point. The main loop of the MAM is organized as follows.

Algorithm 1 (deterministic MAM).

1. Initialization: choose a starting point \mathbf{x}^0 and initial trust region $[\mathbf{a}^0, \mathbf{b}^0]$ such that $\mathbf{x}^0 \in [\mathbf{a}^0, \mathbf{b}^0]$.
2. At the k -th iteration the current approximation to the constrained minimum is \mathbf{x}^k , the current trust region is $[\mathbf{a}^k, \mathbf{b}^k] \subset [\mathbf{a}^0, \mathbf{b}^0]$.
 - (a) Design of Experiments (DoE): a set of points $\mathbf{x}_k^i \in [\mathbf{a}^k, \mathbf{b}^k]$ is chosen to be used for approximation building. Responses are evaluated at the DoE points and approximations are built using the obtained values. Currently, the pool of approximation methods available in MAM consists of a metamodel assemblies¹⁵ and the moving least-squares metamodels¹⁶⁻¹⁹. Other metamodel types could be used as well. Denote the approximate objective function and constraints by $\tilde{F}_0(\mathbf{x})$ and $\tilde{F}_j(\mathbf{x})$, respectively.
 - (b) The original optimization problem (1) is replaced by the following problem:

$$\begin{aligned} & \min_{a_i^k \leq x_i^k \leq b_i^k} \tilde{F}_0(\mathbf{x}) \\ & \text{s.t. } \tilde{F}_j(\mathbf{x}) \leq 1, \quad j = 1, \dots, M, \end{aligned} \quad (2)$$

The approximate problem (2) is solved using Sequential Quadratic Programming (SQP) and the solution of this problem determines the centre of the next trust region.

- (c) The size of the next trust region is determined depending on the quality of approximations at the previous iteration, on the history of the points \mathbf{x}^k , and on the size of the current trust region¹⁰.
- (d) The termination criterion is checked (it is a part of the trust region strategy and depends on the position of the point \mathbf{x}^{k+1} in the current trust region, the size of the current trust region and the quality of approximations). If the termination criterion is satisfied, the algorithm proceeds to step 3. Otherwise, it returns to step 2.
3. Optimization terminates. The obtained approximation to the solution of the problem (1) is \mathbf{x}^{k+1} .

3 Optimization Under Uncertainty

The presence of uncertainties in the responses requires certain reformulations in the problem statement (1). Informally, the optimization problem with uncertainty can be written as follows:

$$\begin{aligned} & \min_{a_i \leq x_i \leq b_i} F_0(\mathbf{x} + \mathbf{u}, \mathbf{y}) \\ & \text{s.t. } F_j(\mathbf{x} + \mathbf{u}, \mathbf{y}) \leq 1, \quad j = 1, \dots, M, \end{aligned} \quad (3)$$

where the random variable \mathbf{u} represents (additive) noise in the design variables and \mathbf{y} is the environmental variable, representing input to the responses that cannot be influenced by the designer. This problem statement is not precise since the random responses $F_0(\mathbf{x} + \mathbf{u}, \mathbf{y})$, $F_j(\mathbf{x} + \mathbf{u}, \mathbf{y})$ cannot be directly optimized. Therefore, a risk measure R needs to be applied to the objective and the constraints:

$$\begin{aligned} & \min_{a_i \leq x_i \leq b_i} R_{u,y}(F_0)(\mathbf{x}) \\ & \text{s.t. } R_{u,y}(F_j)(\mathbf{x}) \leq 1, \quad j = 1, \dots, M, \end{aligned} \quad (4)$$

where the subscript in $R_{u,y}$ indicates that the variables \mathbf{u} and \mathbf{y} ‘collapse’ when the risk measure is applied and the result only depends on \mathbf{x} . The choices of risk measure will be discussed in Section 4. Here the changes in the Algorithm 1 that are necessary to solve the problem (4) are discussed.

First, it needs to be decided on whether to build metamodels for the risk measure $R_{u,y}(F_j)(\mathbf{x})$ or for the original responses $F_j(\mathbf{x} + \mathbf{u}, \mathbf{y})$. The first option may seem attractive due to the simplicity of its implementation in Algorithm 1 but the deciding argument has to be that of the computational cost associated with the evaluation of either function. On the one hand, one evaluation of the function $R_{u,y}(F_j)(\mathbf{x})$ involves computing one or several integrals of the original response $F_j(\mathbf{x} + \mathbf{u}, \mathbf{y})$ over the distribution of \mathbf{u} and \mathbf{y} , which requires many evaluations of the original response. On the other hand, the dimensionality of the argument of $R_{u,y}(F_j)(\mathbf{x})$ is less than that of the argument of $F_j(\mathbf{x} + \mathbf{u}, \mathbf{y})$, therefore, in principle, fewer function evaluations are needed to approximate the former than the latter. In practice, this may be a matter of a trade-off between the two abovementioned effects that depends on the proportion between the numbers of design variables n and the environmental variables n_e and on the complexity of the integration method and metamodeling method used. If the number of design variables is significantly larger than the number of environmental variables, which is often the case and which is assumed in this paper, building metamodels for the original responses $F_j(\mathbf{x} + \mathbf{u}, \mathbf{y})$ is preferred.

This choice requires some changes in the MAM algorithm, since metamodels are now built in a larger space than the one where optimisation is performed in. The changes are summarised below.

Algorithm 2 (MAM under uncertainty).

1. Initialization: choose a starting point $\mathbf{x}_0 \in \mathbb{R}^n$. Choose an initial trust region $[\mathbf{a}^0, \mathbf{b}^0]$ in the combined space of design variables and environmental variables. Ensure that the initial guess \mathbf{x}_0 belongs to the projection of the initial trust region onto the subspace of design variables. The projection of the initial trust region onto the subspace of environmental variables must cover the region where the realisations of the random variable \mathbf{y} are likely to be. The coordinates of the trust region corresponding to the environmental variables will not be changed in the course of optimisation. Typically, the trust region along the coordinates corresponding to

the i -th environmental variable is centred at the mean of the i -th environmental variable and has the width of six of its standard deviations.

2. At the k -th iteration the current approximation to the constrained minimum is \mathbf{x}^k , the current trust region is $[\mathbf{a}^k, \mathbf{b}^k]$.

- (a) Design of Experiments (DoE): a set of points $(x_i^k, y_i^k) \in [\mathbf{a}^k, \mathbf{b}^k]$ is chosen that will be used to build approximations. Responses are evaluated at the DoE points and approximations are built using the obtained values. Denote the approximate objective function and constraints by $\tilde{F}_0(\mathbf{x}, \mathbf{y})$ and $\tilde{F}_j(\mathbf{x}, \mathbf{y})$, respectively.
- (b) The original optimization problem is replaced by the following problem:

$$\begin{aligned} & \min_{a_i \leq x_i \leq b_i} R_{u,y}(\tilde{F}_0)(\mathbf{x}) \\ & \text{s.t. } R_{u,y}(\tilde{F}_j)(\mathbf{x}) \leq 1, \quad j = 1, \dots, M, \end{aligned} \quad (5)$$

The risk measures are calculated using the metamodels obtained in the previous step. Integrals that arise during this process are calculated with respect to the joint distribution of \mathbf{u} and \mathbf{y} . This distribution is assumed to be fixed. The approximate problem (5) is solved using Sequential Quadratic Programming (SQP) and the centre of the next trust region \mathbf{x}^{k+1} is determined as the solution of this problem. The optimisation runs in a subspace of the space where metamodels are built: metamodels are defined in the combined space of design variables and environmental variables, whilst optimisation runs in the subspace of the design variables.

- (c) The trust region is updated. The size of the trust region along the dimensions corresponding to the environmental variables is unchanged. The size of the next trust region along the dimensions corresponding to the design variables is determined depending on the quality of approximations at the previous iteration, on the history of the points \mathbf{x}^k , and on the size of the current trust region according to the trust region strategy explained in Ref. 10.
- (d) The termination criterion is checked. If the termination criterion is satisfied, the algorithm proceeds to step 3. Otherwise, it returns to step 2
3. Optimization terminates. The obtained approximation to the optimum is \mathbf{x}^{k+1} .

Summarising, the key changes introduced to account for uncertainties are (i) metamodels built in the combined space of design variables and environmental variables; (ii) optimisation performed in a subspace of the space where metamodels are defined; (iii) additive noise in the design variables; (iv) application of risk measures to transform random responses into deterministic functions.

4 Risk Measures

As it has been already said, a risk measure is a mapping that transforms a random response $F(\mathbf{x} + \mathbf{u}, \mathbf{y})$ into a deterministic function that can be optimized. Next we provide some examples of risk measures^{2,3}.

- $R(F) = \mu(F)$

5 Conclusion

Recent developments in the Multipoint Approximation Method made it capable of solving large scale problems with uncertainty in parameters modelled as additional ('environmental') random variables and uncertainty in the design variables modelled as additive noise. The approach taken combines metamodels built in the combined space of design variables and environmental variables, application of risk measures that map random responses to deterministic functions, and optimisation in a subspace of the space where metamodels are defined, all within the iterative and trust-region-based framework of MAM. Performance is demonstrated on a benchmark example of structural optimisation known as the scalable cantilevered beam.

Acknowledgement

The authors are grateful for the support provided by the Russian Science Foundation, project No. 16-11-10150.

References

- [1] Back, T. and Schwefel, H.-P. Evolutionary algorithms: Some very old strategies for optimization and adaptation. In *New Computing Techniques in Physics Research II*, Perret-Gallix, D., editor (World Scientific, Singapore, 1992).
- [2] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on* **6**(2), 182–197 Apr (2002).