A probabilistic performance-oriented control gain optimization approach with rare-event chance constraints is introduced for flight systems. Aiming at estimating rare probabilities accurately and efficiently, subset simulation is combined with a global surrogate to improve efficiency. At each level of subset simulation, the samples that are close to the failure domain are employed to construct a local surrogate model, and the global surrogate is thereby refined progressively. In return, seed candidates are screened by the global surrogate, thus saving a large number of calls to the true model and reducing computational expense. Then, control parameters are optimized under chance constraints to directly guarantee system performance. Simulations are conducted on an aircraft longitudinal model subjecting to parametric uncertainties to demonstrate the efficiency and accuracy of this method.

I. Introduction

Various uncertainties and disturbances are inevitable in real flight and might endanger the safety of aircraft. In order to guarantee flight safety, safety requirements in flight control systems must be fulfilled with the existence of uncertainties and disturbances. The safety requirements are defined as failure probability thresholds that the system must not exceed. For example, during automatic landing, the probabilities of exceeding safety limits must not be larger than the thresholds given in [1]. The acceptable failure probability thresholds are usually quite small ($10^{-9}$–$10^{-5}$), especially for failure events that may lead to severe safety issues [1–3], which makes it even harder to choose reasonable control parameters.

Subset simulation [2,4,5] is an efficient method for estimating rare failure probabilities where the probability of a rare event is expressed as a product of much larger conditional probabilities. Although it achieves higher efficiency than Monte Carlo simulation (MCS), at least thousands of evaluations of the original model are required to reach a desirable accuracy [6]. This is acceptable when estimating the probability of rare events, but for optimization problems with rare-event chance constraints, the computational costs are generally unaffordable.

To enhance the efficiency of subset simulation, many researchers combine it with surrogate models since these are usually analytical and it is very efficient to simulate with such models. One possible scheme is to perform subset simulation with the surrogate model instead of the true model (limit-state function), but this necessitates a sufficiently accurate approximation of the true model. This goal has been successfully achieved by surrogate models such as kriging [2], support vector machines (SVM) [8,9], and neural networks (NN) [10]. In these works, adaptive training strategies that save a large number of calls to the true model are employed to construct highly accurate surrogates for intermediate limit-state surfaces within subset simulation. An alternative scheme is to accelerate subset simulation with the surrogate model. For example, a delayed rejection strategy where the samples are first screened by the surrogate model is introduced during the Markov chain Monte Carlo (MCMC) sampling [6]. Although the unbiasedness of this algorithm is guaranteed, the computational and statistical efficiency rely on the quality of the surrogate model.

Chance constraints have been frequently used in control community. The results in [11] introduce both analytical methods and sampling-based methods to handle chance constrains, and the feedback controller is incorporated into the optimization with a risk allocation. A strategy based on split Bernstein polynomials and MCMC is implemented to estimate chance constraints for optimal control problems in [12]. In [13], polynomial chaos expansion and subset simulation are employed to approximate rare-event probabilities within a chance-constrained open-loop optimal control framework. Besides, chance constraints are also common in model predictive control (MPC) [14]. Min-max theory is a very popular method for robust MPC and it tries to achieve a worst-case design to increase robustness [15,16]. Chance constraints are converted into explicit algebraic constraints to assure the online applicability of MPC in [17]. In addition, polynomial chaos expansion is utilized to propagate the parametric uncertainties and probabilistic constraints.

---

*Ph.D. Candidate, dalong.shi@tum.de.
†Professor, Associate Fellow AIAA, florian.holzapfel@tum.de.
are transformed into the second-order cone constraints in [18]. Among these research, only the study [13] is specifically tailored to rare failure probabilities, but the surrogate model in the failure domain might not be accurate enough.

This paper develops an effective strategy to accelerate the estimation of rare failure probabilities. The strategy is then applied to increase the efficiency of flight control optimization which directly ensures the satisfaction of rare-event chance constraints. In this approach, a local response surface-based surrogate for the true model is constructed using the samples close to the failure domain at each level of subset simulation. The global surrogate is then updated by the local surrogate. Afterwards, a substantial proportion of calls to the true model are substituted by the calls to the global surrogate.

The rest of the paper is organized as follows. Section II recalls the basics of subset simulation, polynomial chaos expansion and response surface method. Section III introduces surrogate-based subset simulation so as to progressively refine the surrogate model and accelerate subset simulation. Section IV presents the framework of performance-guaranteed control optimization with rare-event chance constraints. In Section V, the proposed framework is implemented on a flight control optimization problem with rare-event probabilistic requirements.

II. Basics on Subset Simulation and Surrogate Model Construction

A. Subset Simulation

The failure probability of a certain event can be generally expressed as follows:

$$p_f = \int_F \rho(\theta)d\theta,$$

where \( \theta = [\theta_1, \theta_2, \ldots, \theta_p]^T \in \mathbb{R}^p \) is a vector of uncertain variables, \( \rho(\theta) \) is the joint probability density function (PDF) of \( \theta \), and \( F = \{ \theta \in \mathbb{R}^p : s(\theta) \leq 0 \} \) is the failure domain in which \( s(\theta) \) is the so-called limit-state function.

The key idea of subset simulation (SuS) is to introduce \( m \) intermediate failure domains \( F_j, j = 1, \ldots, m \) satisfying \( F_1 \supset F_2 \supset \cdots \supset F_m = F \), such that the failure probability of a rare event is transcribed into a product of conditional probabilities [4, 5]:

$$p_f = \mathbb{P}[F_m] = \mathbb{P}[F_m|F_{m-1}] \mathbb{P}[F_{m-1}] = \cdots = \prod_{j=1}^{m} \mathbb{P}[F_j|F_{j-1}],$$

where \( \mathbb{P}[\cdot] \) is the probability operator. \( F_0 \) is the uncertain parameter space and \( \mathbb{P}[F_j|F_0] = \mathbb{P}[F_1] \). In practice, intermediate failure domains are denoted as \( F_j = \{ \theta \in \mathbb{R}^p : s(\theta) \leq b_j \} \) with \( b_1 > b_2 > \cdots > b_m = 0 \), where \( b_j, j = 1, \ldots, m \) are the thresholds of \( s(\theta) \). Thresholds \( b_j \), \( j = 1, \ldots, m-1 \) are selected reasonably such that the conditional probabilities \( p_j = \mathbb{P}[F_j|F_{j-1}], j = 1, \ldots, m \) are large enough (\( p_j \in [0.1, 0.3] \) is suggested for the best performance [19]) to be efficiently estimated by simulations.

The basic SuS algorithm can be summarized as given in Algorithm 1. Various MCMC sampling strategies have been developed to enhance the performance of SuS, such as the modified (component-wise) Metropolis-Hastings algorithm [4], the modified Metropolis-Hastings algorithm with delayed rejection [20] and the Gaussian conditional sampling [21]. In this paper, the Gaussian conditional sampling is applied, using which the candidate samples generated from the proposal PDF always differ from the current sample. Therefore, the number of samples generated at each subset level (\( j = 1, \ldots, m-1 \)) is

$$N + (m-1)(1 - p_0)N,$$

and the total number of calls to the true model is

$$N + (m - 1)(1 - p_0)N.$$

The coefficient of variation (c.o.v.) of the estimated failure probability can be evaluated by the c.o.v. of the estimated intermediate conditional probabilities, see [4] for details.

B. Polynomial Chaos Expansion

Consider a system described by the limit-state function \( s(\theta) \). Its polynomial chaos expansion (PCE) can be represented as an infinite weighted sum of polynomial bases [22, 23]:

$$s(\xi) = \sum_{i=0}^{\infty} a_i \Psi_i(\xi),$$

where...
where $\xi = [\xi_1, \xi_2, \ldots, \xi_p]^T \in \mathbb{R}^p$ represents a vector of standard random variables, $\Psi_i(\xi)$ the multivariate polynomial basis functions, and $a_i$ the corresponding expansion coefficients. In practical problems, $\theta$ is usually not a vector of standard random variables, therefore it is necessary to transform $\theta$ into a set of standard variables $\xi$ through the isoprobabilistic transformation:

$$\xi = \tau(\theta).$$

The multivariate polynomials $\Psi_i(\xi)$ can be constructed as the product of their univariate counterparts:

$$\Psi_i(\xi) = \prod_{r=1}^p \psi_{m^r_i}(\xi_r),$$

where $m^r_i$ denote the multi-indexes that contain all the possible combinations of univariate polynomials and $\psi_{m^r_i}(\xi_r)$ denote the $m^r_i$th-degree univariate orthogonal polynomial bases that satisfy

$$\mathbb{E}[\psi_i(\xi)\psi_j(\xi)] = \int_\Omega \psi_i(\xi)\psi_j(\xi)\rho(\xi)d\xi = \gamma_{ij}\delta_{ij}, \quad i, j \in \mathbb{N},$$

in which $\mathbb{E}[\cdot]$ is the expectation with respect to $\rho(\xi)$, $\Omega$ is the support of $\xi$, $\gamma_i = \mathbb{E}[\psi_i^2(\xi)]$, and $\delta_{ij}$ is the Kronecker function equal to 1 when $i = j$ and 0 otherwise. Consequently, the orthogonality also holds for $\Psi_i(\xi)$:

$$\mathbb{E}[\Psi_i(\xi)\Psi_j(\xi)] = \gamma_{ij}\delta_{ij}, \quad i, j \in \mathbb{N},$$

where $\gamma_i = \mathbb{E}[\Psi_i^2(\xi)]$.

The polynomial bases $\psi_i$ can be selected from Table I based on the distribution of $\xi$. Table I shows classical families of orthogonal polynomials in the Wiener-Askey scheme.

The expansion with infinite terms in Eq. (5) is exact, but in practice, only a finite number of terms can be computed. Usually, the expansion is truncated up to a certain degree $M$ and the limit-state function is approximated by the surrogate model $\tilde{s}$ built by the $M$-th order PCE:

$$s(\xi) \approx \tilde{s}(\xi) = \sum_{i=0}^{P-1} a_i\Psi_i(\xi),$$

where the number of terms is generally given by

$$P = \frac{(M + p)!}{M!p!}.$$
As a result, the residual error $\epsilon$ is still required to achieve a sufficient accuracy. Several methods [7–10] have been proposed to build highly accurate model evaluations.

A rule of thumb for designating the size of experimental design is
\[
A \quad \text{are still required to achieve a sufficient accuracy.}
\]

### A. Subset Simulation Based on Response Surface Method

Even though SuS is very efficient in estimating rare-event probabilities, at least thousands of calls to the true model are still required to achieve a sufficient accuracy. Several methods [7–10] have been proposed to build highly accurate model evaluations.

#### III. Surrogate-Based Subset Simulation

**C. Response Surface Method**

Response surface method (RSM) [26] is a useful tool to establish the relationship between input variables and the model response. Generally, the relationship, i.e., response surface or surrogate model, is assumed to be a low-order polynomial with unknown coefficients and the coefficients can be obtained by least squares regression. This is similar to PCE using regression method, and the latter can be regarded as an extension of the former to the standard random space. In RSM, the orthogonality as shown in Eq. (9) is not fulfilled, but one can still employ the framework in Eq. (12) to achieve the response surface construction.

### 1. Least squares regression

Least squares regression aims at finding a set of coefficients $a = [a_0, a_1, \ldots, a_P]^T$ that minimize the mean square residual error with a collection of samples (experimental design) \{\(\xi^{(i)}: i = 1, 2, \ldots, n_s\}\} of size $n_s$ and the associated model evaluations $y = [s(\xi^{(1)}), s(\xi^{(2)}), \ldots, s(\xi^{(n_s)})]^T$ [24]:
\[
a = \arg \min A^T \mathbb{E} [ (s(\xi) - \tilde{s}(\xi))^2 ] = (A^T A)^{-1} A^T y,
\]
where $A$ is the experimental matrix with elements
\[
A_{ij} = \Psi_{j-1}(\xi^{(i)}), \quad i = 1, \ldots, n_s; j = 1, \ldots, P.
\]
A rule of thumb for designating the size of experimental design is $n_s \approx 2P–3P$ [24].

#### 2. Spectral projection

Due to the orthogonality of polynomial bases, one can project the random response in Eq. (10) against each basis and obtain the expansion coefficients [25]:
\[
a_i = \frac{1}{\gamma_i} \mathbb{E} [ s(\xi) \Psi_i(\xi) ] = \frac{1}{\gamma_i} \int_{\Omega} s(\xi) \Psi_i(\xi) \rho(\xi) d\xi.
\]
As a result, the residual error $\epsilon_r = s(\xi) - \tilde{s}(\xi)$ is orthogonal to the chosen bases. The integral in Eq. (14) can be evaluated numerically by Gaussian quadrature or sparse quadrature, which is a weighted-sum scheme:
\[
a_i \approx \frac{1}{\gamma_i} \sum_{j=1}^{Q} s(\xi^{(j)}) \Psi_i(\xi^{(j)}) w^{(j)},
\]
where $\xi^{(j)}$ are the quadrature nodes, $w^{(j)}$ are the corresponding weights, and $Q$ is the number of these quadrature nodes.

### Table 1 Classical families of orthogonal polynomials [22].

| Type of variable $\xi$ | Distribution $\rho(\xi)$ | Support $\Omega$ | Orthogonal polynomial $\psi_n(\xi)$ |
|------------------------|--------------------------|-----------------|----------------------------------|
| Gaussian               | $\frac{1}{\sqrt{2\pi}} e^{-\xi^2/2}$ | $(-\infty, \infty)$ | Hermite $H_n(\xi)$ |
| Uniform                | $\frac{1}{2}$ | $[-1, 1]$ | Legendre $P_n(\xi)$ |
| Gamma                  | $\xi^a e^{-\xi} \Gamma(a+1)$ | $[0, \infty)$ | Laguerre $L_n^a(\xi)$ |
| Beta                   | $\frac{(1-\xi)^{a}(1+\xi)^{b}}{2^{a+b}B(a+1,b+1)}$ | $[-1, 1]$ | Jacobi $J_n^a(\xi)$ |

Several non-intrusive methods have been developed to compute the expansion coefficients, among which least squares regression and spectral projection are the most fundamental and widely used techniques.
surrogates for intermediate limit-state surfaces \( \theta \in \mathbb{R}^p : s(\theta) = b_j \). Alternatively, surrogate-based subset simulation (SBSS) is proposed to refine the surrogate of the limit-state function \( s(\theta) \) in the domain of interest and accelerate the conventional SuS with the updated surrogate model.

To start with, we assume that the surrogate model is sufficiently accurate. At each level \( j = 0, 1, \ldots, m - 1 \) of SuS, samples \( \{ \theta_j^{(i)} : i = 1, \ldots, N \} \) are generated and the true model are called for \( N \) or \( (1 - p_0)N \) times. Since the surrogate model is analytical and can be evaluated at almost no cost, fewer points are preferably chosen to be evaluated with the true model in SBSS. First, the limit-state values of the \( N \) samples are estimated using the surrogate model \( \tilde{s}_j(\theta) \). Second, these values are sorted in increasing order and the intermediate failure domain \( \tilde{F}_{j+1} \) is set as \( \tilde{F}_{j+1} = \{ \theta \in \mathbb{R}^p : \tilde{s}_j(\theta) \leq \tilde{b}_{j+1} \} \), where \( \tilde{b}_{j+1} \) is the \( p_0 \)-percentile (\( p_0 \leq 1 \)) of these estimated limit-state values. The number of samples \( \theta_j^{(i)} \in \tilde{F}_{j+1} \) is given by \( \bar{N}_s \). Third, the limit-state values of \( \theta_j^{(i)} \in \tilde{F}_{j+1} \) are computed with the true model \( s(\theta) \). Last, these values are sorted in ascending order and the intermediate failure domain \( F_{j+1} \) is set as \( F_{j+1} = \{ \theta \in \mathbb{R}^p : s(\theta) \leq b_{j+1} \} \), where \( b_{j+1} \) is the \( p_0 \)-percentile of the limit-state values. The samples \( \theta_j^{(i)} \in F_{j+1} \) are chosen as seeds \( \{ \theta_{j,seed}^{(i)} : i = 1, \ldots, N_s \} \), where \( N_s = p_0N \). Different categories of samples and intermediate limit-state surfaces are depicted in Fig. 1.

The procedure above includes two steps of selecting points according to the sorted function values, of which the former is called pre-selection. The aim of pre-selection is to narrow the possible range of seeds to \( \tilde{F}_{j+1} \) such that it is not necessary to evaluate the true model beyond the reduced scope. As a result, the number of calls to the true model decreases to \( \bar{N}_s \). Note that the seeds can be chosen without any error if \( \bar{N}_s \) (or \( \tilde{b}_{j+1} \)) is large enough. The margin between \( \{ \theta \in \mathbb{R}^p : s(\theta) = \tilde{b}_{j+1} \} \) and \( \{ \theta \in \mathbb{R}^p : s(\theta) = b_{j+1} \} \), as shown in Fig. 1, allows limited errors of the surrogate model, and the percentage \( p_0 \) can be assigned depending on the quality of the surrogate model.

Besides the pre-selection step, the surrogate model is also utilized in MCMC sampling when the generated samples are decided to be accepted or rejected. This indicates that no additional calls to the true model are required here, but the surrogate precision around the threshold \( b_{j+1} \) should be guaranteed.

Next, we refine the surrogate model progressively as SuS is performed level by level. In the initial level of SuS where the crude MCS is implemented, an initial global surrogate model \( \tilde{s}_0(\theta) \) is constructed by PCE. Whereas in other levels, RSM is applied to build local surrogates \( \tilde{s}_{j,local}(\theta), j = 1, \ldots, m - 1 \). At each level, take the samples \( \theta_j^{(i)} \in \tilde{F}_{j+1} \) as the experimental design (ED) for RSM \( \{ \theta_j^{(i)} : i = 1, \ldots, \bar{N}_s \} \). The obtained local surrogate \( \tilde{s}_{j,local}(\theta) \) is only valid within the sample range. Hence, it is combined with previous knowledge of the system to update the global surrogate:

\[
\tilde{s}_j(\theta) = \begin{cases}
\tilde{s}_{j-1}(\theta), & \tilde{s}_{j-1}(\theta) \geq \tilde{b}_j; \\
\tilde{s}_{j,local}(\theta), & \tilde{s}_{j-1}(\theta) < \tilde{b}_j.
\end{cases}
\]

In this formulation, the global surrogate is expressed as a piecewise function with jump discontinuities between subdomains. The influence of the discontinuities in \( F_j \) can be alleviated by selecting a larger \( \tilde{b}_j \), but this increases the number of calls to the true model. It is thereby a tradeoff between accuracy and computational expense.

In summary, the proposed strategy refines the global surrogate model gradually as the samples approach the failure domain, and in return a significant number of evaluations of the true model are approximated by calling the surrogate model. A detailed algorithm is given in Algorithm 2. It is beneficial, especially when the evaluation of the true model is
computationally demanding. The number of calls to the true model at each subset level is

\[ \tilde{p}_0 N, \]

and the total number of calls is

\[ N_0 + (m - 1)\tilde{p}_0 N, \]

where \( N_0 \) is the number of true model evaluations when building the initial surrogate \( \tilde{s}_0(\theta) \) by PCE.

**Algorithm 2** Basic algorithm for surrogate-based subset simulation

**Input:** Limit-state function \( s(\theta) \); The distribution of uncertain variables \( \rho(\theta) \); The number of samples at each level \( N \); Conditional probability \( p_0 \); The percentage of samples for the construction of local surrogates \( \tilde{p}_0 \) (\( \tilde{p}_0 \geq p_0 \)).

**Output:** The estimation of failure probability \( \tilde{p}_f \).

1. Generate \( N \) independent and identically distributed (i.i.d.) samples \( \{\theta^{(i)}_0 : i = 1, \ldots, N\} \) according to \( \rho(\theta) \) and construct a global surrogate model of limit-state function \( \tilde{s}_0(\theta) \) by PCE;
2. Calculate the limit-state values \( \tilde{s}_0(\theta^{(i)}_0) : i = 1, \ldots, N \); 
3. Sort these values in ascending order, find the \( \tilde{p}_0 \)-percentile \( \tilde{b}_1 \), and set \( \tilde{F}_1 = \{\theta \in \mathbb{R}^p : \tilde{s}_0(\theta) \leq \tilde{b}_1\} \);
4. Select samples \( \theta^{(i)}_0 \in \tilde{F}_1 \) as the experimental design \( \{\theta^{(i)}_{0,ED} : i = 1, \ldots, \tilde{N}_s\} \), where \( \tilde{N}_s = \tilde{p}_0 N \);
5. Calculate the limit-state values \( s(\theta^{(i)}_{0,ED}) : i = 1, \ldots, \tilde{N}_s \);
6. Sort these limit-state values in ascending order, find the \( p_0 \)-percentile \( b_1 \), and set \( F_1 = \{\theta \in \mathbb{R}^p : s(\theta) \leq b_1\} \);
7. Set \( j = 1 \);
8. while \( b_j > 0 \) do
9. Build a local surrogate \( \tilde{s}_{j,local}(\theta) \) implementing RSM with the experimental design \( \{\theta^{(i)}_{j-1,ED} : i = 1, \ldots, \tilde{N}_s\} \) and the corresponding limit-state values \( \{s(\theta^{(i)}_{j-1,ED}) : i = 1, \ldots, \tilde{N}_s\} \);
10. Refine the current global surrogate model \( \tilde{s}_{j-1}(\theta) \) to be \( \tilde{s}_j(\theta) \) as in Eq. (16);
11. Choose samples \( \theta^{(i)}_{j-1} \in F_j \) as seeds \( \{\theta^{(i)}_{j-1,seed} : i = 1, \ldots, \tilde{N}_s\} \), where \( \tilde{N}_s = p_0 N \);
12. Generate \( N \) samples \( \{\theta^{(i)}_j : i = 1, \ldots, N\} \) from the seeds applying MCMC sampling, wherein samples are accepted or rejected using the updated global surrogate model \( \tilde{s}_j(\theta) \);
13. Calculate the limit-state values \( \tilde{s}_j(\theta^{(i)}_j) : i = 1, \ldots, N \);
14. Sort these values in ascending order, find the \( \tilde{p}_0 \)-percentile \( \tilde{b}_{j+1} \), and set \( \tilde{F}_{j+1} = \{\theta \in \mathbb{R}^p : \tilde{s}_j(\theta) \leq \tilde{b}_{j+1}\} \);
15. Select samples \( \theta^{(i)}_j \in \tilde{F}_{j+1} \) as the experimental design \( \{\theta^{(i)}_{j,ED} : i = 1, \ldots, \tilde{N}_s\} \);
16. Calculate the limit-state values \( \{s(\theta^{(i)}_{j,ED}) : i = 1, \ldots, \tilde{N}_s\} \);
17. Sort these limit-state values in ascending order, find the \( p_0 \)-percentile \( b_{j+1} \), and set \( F_{j+1} = \{\theta \in \mathbb{R}^p : s(\theta) \leq b_{j+1}\} \);
18. Set \( j = j + 1 \);
19. end while
20. Obtain the number of subsets \( m = j \);
21. Identify the number of samples \( \theta^{(i)}_{m-1} \in F_j : N_F \);
22. Estimate the failure probability \( \tilde{p}_f = p_0^{m-1} \frac{N_F}{N} \);
23. return \( \tilde{p}_f \).

**B. Adaptive Response Surface Method**

As one local surrogate model is built at each level of SBSS, any surrogate could influence the precision of SBSS. The quality of the constructed surrogate model can be measured by the mean square error of the residual (i.e. empirical error):

\[ \epsilon_{emp} = \frac{1}{N_s} \sum_{i=1}^{N_s} \left( s(\theta^{(i)}) - \tilde{s}(\theta^{(i)}) \right)^2. \]

(19)

Its normalized quantity, which is called relative empirical error, is computed by

\[ \epsilon^*_{emp} = \frac{\epsilon_{emp}}{\text{var}[y]} \]

(20)
where \( \text{var}[y] \) is the variance of responses \( y \). The error is reduced with the increase of the order of RSM until the response surface fits the samples in experimental design perfectly, i.e., \( \epsilon_{\text{emp}} \) or \( \epsilon_{\text{emp}} \) is almost zero. However, with sufficiently high order, the risk involved is that the approximation of these samples can be extremely good but very bad elsewhere. This situation is known as overfitting, which means the response surface would be quite different with that built by another set of samples. Therefore, the surrogate modeling error is usually underestimated by the empirical error, but can be better estimated by the leave-one-out (LOO) cross-validation error \([24]\). Leave-one-out cross-validation removes one point \( \theta^{(i)} \) of the experimental design and construct a surrogate model denoted by \( \tilde{s}^{(-i)}(\theta) \) from the remaining \( n_s - 1 \) samples. The predicted residual error at \( \theta^{(i)} \) reads \([24]\):

\[
\epsilon_i = s(\theta^{(i)}) - \tilde{s}^{(-i)}(\theta^{(i)}) = \frac{s(\theta^{(i)}) - \tilde{s}(\theta^{(i)})}{1 - h_i},
\]

where \( h_i \) is the \( i \)-th diagonal term of matrix \( A(A^T A)^{-1}A^T \). The leave-one-out error is defined as

\[
\epsilon_{\text{LOO}} = \frac{1}{n_s} \sum_{i=1}^{n_s} \epsilon_i^2 = \frac{1}{n_s} \sum_{i=1}^{n_s} \left( \frac{s(\theta^{(i)}) - \tilde{s}(\theta^{(i)})}{1 - h_i} \right)^2.
\]

Similarly, the relative leave-one-out error can be obtained by

\[
\epsilon'_{\text{LOO}} = \frac{\epsilon_{\text{LOO}}}{\text{var}[y]}.
\]

The adaptive RSM that minimizes the LOO error is summarized in Algorithm 3. By this means, excessively high orders are prohibited and the overfitting problem is avoided.

**Algorithm 3 Adaptive RSM algorithm**

**Input:** Limit-state function \( s(\theta) \); The number of uncertain parameters \( p \); The possible orders of the response surface \( M_{\text{min}} : M_{\text{max}} \); The size of experimental design \( n_s \).

**Output:** The optimal surrogate model \( \tilde{s}(\theta) \)

1. Obtain the experimental design \( \{\theta^{(i)} : i = 1, \ldots, n_s\} \);
2. Calculate the corresponding limit-state values \( \{s(\theta^{(i)}) : i = 1, \ldots, n_s\} \);
3. for \( M = M_{\text{min}} : M_{\text{max}} \) do
   4. Generate polynomial bases \( \{\Psi_i(\theta) : i = 0, 1, \ldots, P - 1\} \);
   5. Calculate the experimental matrix \( A \) as in Eq. (13);
   6. Solve the least squares problem in Eq. (12);
   7. Compute \( \epsilon'_{\text{LOO}}(M) \) according to Eqs. (22) and (23);
4. end for
9. \( M^* = \arg \min \epsilon'_{\text{LOO}}(M) \);
10. Obtain the optimal surrogate model \( \tilde{s}(\theta) \);
11. return \( \tilde{s}(\theta) \).

**IV. Performance-Guaranteed Control Optimization**

Consider a class of closed-loop dynamic systems subject to parametric uncertainties:

\[
\begin{align*}
\dot{x}(t) &= f(x(t), u(t), \theta), \\
y(t) &= g(x(t), u(t), \theta),
\end{align*}
\]

where \( x \in \mathbb{R}^{n_x} \) represents the state vector, \( u \in \mathbb{R}^{n_u} \) the input vector, and \( y \in \mathbb{R}^{n_y} \) the output vector. The vector \( k = [k_1, k_2, \ldots, k_d]^T \in \mathbb{R}^d \) contains control gains that are uncorrelated with uncertain parameters \( \theta \). The functions \( f(\cdot) \) and \( g(\cdot) \) describe the system dynamics and are assumed to be known. The performance of the dynamic system can be depicted by a series of performance functions \( h_i(k, \theta), i = 0, 1, \ldots, n_h \). These functions can be implicit, such as settling time, overshoot, and stability margins.
The control optimization that guarantees system performance is stated as follows:

\[
\min_k \mathbb{P}[h_0(k, \theta) < C_0],
\]

\[
\text{s.t.} \quad \mathbb{P}[h_i(k, \theta) < C_i] < \beta_i, \quad i = 1, \ldots, n_h, \quad c_i(k) \leq 0, \quad i = 1, \ldots, n_c, \tag{25}
\]

where \( C_i, i = 0, 1, \ldots, n_h \) are thresholds for these performance functions, \( \beta_i, i = 1, \ldots, n_h \) are the probabilistic bounds that should not be violated, and \( c_i(k), i = 1, \ldots, n_c \) are deterministic constraint functions. This control optimization framework aims at minimizing the exceeding probability of \( h_0(k) \) while ensuring that other exceeding probabilities of performance functions are within their corresponding safe ranges. Consequently, it is able to directly fulfill the statistical requirements and explore further performance.

V. Application to Flight Control

A. Simulation Model

The control plant is an aircraft (Diamond DA42) longitudinal model integrating actuator dynamics, structural mode, and filters. A proportional-integral-derivative (PID) controller with feedforward control is implemented:

\[
\dot{q}_{\text{cmd}} = k_H n_{z,\text{cmd}} + k_{n_z} n_z + k_I \int (n_{z,\text{cmd}} - n_z) \, dt + k_q \omega_y, \tag{26}
\]

where \( \dot{q}_{\text{cmd}} \) is the pitch acceleration command, \( \omega_y \) is the pitch angular rate, and \( n_z \) and \( n_{z,\text{cmd}} \) are the vertical load factor and its command. \( k = [k_H, k_{n_z}, k_I, k_q]^T \) is the vector of control gains.

Uncertainties in aerodynamic derivatives are considered and the relative values with regard to the reference values are assumed to be Gaussian-distributed:

\[
\begin{bmatrix}
M_\alpha/M_{\alpha,\text{ref}} \\
M_q/M_{q,\text{ref}} \\
M_\eta/M_{\eta,\text{ref}}
\end{bmatrix}
\sim N\left(
\begin{bmatrix}
\mu_\alpha \\
\mu_q \\
\mu_\eta
\end{bmatrix}
\left[
\begin{bmatrix}
\sigma^2_\alpha \\
\sigma^2_q \\
\sigma^2_\eta
\end{bmatrix}
\right]
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}
\right),
\tag{27}
\]

where \( M_\alpha, M_q, \) and \( M_\eta \) are the aerodynamic moments about the angle of attack \( \alpha \), the pitch rate \( q \), and the elevator deflection \( \eta \) respectively. The mean \( \mu_\alpha = \mu_q = \mu_\eta = 1 \) and the standard deviation \( \sigma_\alpha = \sigma_q = \sigma_\eta = 0.15 \).

In the end, the closed-loop system is linear and can be represented as

\[
\begin{aligned}
\dot{x} &= A_{cl} x + B_{cl} u, \\
y &= C_{cl} x,
\end{aligned}
\tag{28}
\]

where \( A_{cl}, B_{cl}, \) and \( C_{cl} \) are closed-loop system matrices. The dependence of these system matrices on \( k \) and uncertain aerodynamic derivatives has been omitted to simplify the notation. Besides the state introduced by the integration element of the control structure, the state vector \( x \) contains the states of short period dynamics (\( \alpha \) and \( q \)), actuator dynamics, structural mode, and notch filter. Each of them is modeled as a second-order system. The output \( y = n_z \) and the input vector \( u = [n_{z,\text{cmd}}, w_z]^T \), where \( w_z \) is the vertical wind velocity of the standard “1-cosine” gust [27]:

\[
w_z = \begin{cases}
0, & x_g < 0; \\
\frac{v_g}{2} \left[ 1 - \cos \left( \frac{\pi x_g}{d_g} \right) \right], & 0 \leq x_g < d_g; \\
v_g, & x_g \geq d_g,
\end{cases}
\tag{29}
\]

where \( x_g \) is the traveled horizontal distance, \( d_g \) is the gust length, and \( v_g \) is the gust amplitude. In this application, \( d_g = 91.4 \text{m} \) and \( v_g = 13.9 \text{m/s} \).

B. Rare-Event Probability Estimation

Consider the stability margin requirements [28] as follows:

\[
\mathbb{P}[h_1(\theta) < 6dR] < 10^{-6},
\]

\[
\mathbb{P}[h_2(\theta) < 45^\circ] < 10^{-6},
\tag{30}
\]

8
where $h_1(\theta)$ and $h_2(\theta)$ denote the performance functions of gain margin (GM) and phase margin (PM) respectively given fixed control gains. Note that no analytical form is available for these performance functions. To directly guarantee the compliance with the rare-event constraints, the rare-event probabilities must be precisely estimated first. In the following part, the proposed SBSS is applied to accomplish this task. The initial surrogate model is constructed by the 5th-order PCE (spectral projection) and a tensor product quadrature with 6 nodes in each dimension (216 nodes in total). The Gaussian conditional sampling [21] is implemented in both SuS and SBSS, and other settings are as follows: $N = 2000; \rho_0 = 0.1; \bar{\rho}_0 = 0.11; M_{\min} : M_{\max} = 2 : 7$.

Figure 2 shows the qualities of the surrogate models constructed by RSM of different orders. The two subplots corresponding to $h_1(\theta)$ and $h_2(\theta)$ at a certain level of SBSS illustrate that the empirical error underestimates the fitting errors especially when excessively high orders are adopted. The surrogate model with the minimum LOO error is obtained by the adaptive RSM and the overfitting problem is thus avoided.

The results of SBSS (level 0, 1, \ldots, 5) are given in Figs. 3, 4, and 6. Figure 3 depicts the adopted orders of the adaptive RSM and the LOO errors of the built response surfaces. The subfigures show that the performance function $h_1(\theta)$ can be approximated by a low (3rd or 4th) order polynomial whereas a relatively high (5th or 6th) order polynomial may be required to describe the nonlinearity of $h_2(\theta)$.
In Fig. 4, the responses of the true model (TM) and the surrogate model (SM) are compared given the same uncertain inputs. This comparison is also conducted between the responses of the true model and the surrogate model constructed by the 10th-order PCE (spectral projection) in Fig. 5. In these two figures, the closer these points lie to the line $y = x$, the better the surrogates. The noncompliance with the reference line in Fig. 5b indicates that the performance function $h_2(\theta)$ can hardly be represented by a polynomial even though the order is as high as 10. However, it can be solved by a relatively low order piecewise polynomial surrogate as shown in Fig. 4b. This implies that $h_2(\theta)$ might be highly nonlinear or even not smooth, and the nonlinearity mainly influence the surrogate model at level 0. Despite the poor initial surrogate model, further ones towards the failure domain are constructed successfully. Therefore, building a piecewise surrogate model could be an effective way to approximate a highly nonlinear function. On the contrary, $h_1(\theta)$ can be expressed as a polynomial function since all the samples in Fig. 5a are perfectly close to the line $y = x$.

![Fig. 4 Response comparisons (TM vs. SM by SBSS).](image1)

![Fig. 5 Response comparisons (TM vs. SM by PCE).](image2)

The results of cumulative distribution function (CDF) are depicted in Fig. 6. The samples in SBSS approach the increasingly rare domain level by level, which behave the same as those in SuS. Besides the CDFs, the failure probabilities of events $h_1(\theta) < 7.4dB$ and $h_2(\theta) < 55^\circ$ are also assessed in Fig. 7. With 80 replicate assessments,
Figures 8 and 9 illustrate the estimations of c.o.v. and $p_f$ using different numbers of samples at each simulation level. Here, the c.o.v. bound is the averaged estimates of c.o.v. upper or lower bounds [4] over 80 simulations, whereas the empirical (emp.) c.o.v. is the sample c.o.v. of the failure probability estimates over these independent runs. The 3-$\sigma$ range is considered to cover nearly all the possible failure probabilities. It is estimated by the lower bound of c.o.v. in Fig. 7 and by the emp. c.o.v. in Fig. 9. In these graphs, both the CDFs and the failure probabilities estimated by SBSS are in great accordance with their counterparts by SuS, demonstrating the comparable performance of SBSS. Similar comparisons are conducted for different thresholds (2000 samples at each level of SuS and SBSS) in Figs. 10 and 11, and the matches of the results also prove that SBSS owns the equivalent capability of estimating rare event probabilities.

Fig. 6 The estimations of CDFs using SBSS.

Fig. 7 The estimations of CDFs and failure probabilities.

The numbers of calls to the true model (denoted by $N_{\text{call}}$) using SuS and SBSS are depicted in Tables 2 and 3. Table 2 shows the numbers of true model evaluations with different numbers of samples per level $N$ and Table 3 shows those with different numbers of total levels $m$. In all cases, at least 86% of the calls to the true model are saved in SBSS, demonstrating its superior efficiency over SuS.
Fig. 8  The estimations of coefficients of variation.

Fig. 9  The estimations of failure probabilities.

Fig. 10  The estimations of coefficients of variation.
a) The results of gain margin  
b) The results of phase margin

Fig. 11 The estimations of failure probabilities.

Table 2 The numbers of calls to the true model with different number of samples per level.

| N   | m | \( N_{\text{call}} \) (SuS) | \( N_{\text{call}} \) (SBSS) |
|-----|---|-------------------------------|-------------------------------|
| 1000| 6 | 5500                          | 766                           |
| 2000| 6 | 11000                         | 1316                          |
| 3000| 6 | 16500                         | 1866                          |
| 5000| 6 | 27500                         | 2966                          |

Table 3 The numbers of calls to the true model with different numbers of total levels.

| N   | m | \( N_{\text{call}} \) (SuS) | \( N_{\text{call}} \) (SBSS) |
|-----|---|-------------------------------|-------------------------------|
| 2000| 2 | 3800                          | 436                           |
| 2000| 4 | 7400                          | 876                           |
| 2000| 6 | 11000                         | 1316                          |
| 2000| 8 | 14600                         | 1756                          |
| 2000|10 | 18200                         | 2196                          |

C. Control Parameter Optimization

Besides the stability margin requirements in Eq. (30), tracking performance and disturbance rejection performance are considered as well in the optimization framework Eq. (25). The objective function and the probabilistic constraint functions are defined in Table 4.

Table 4 The objective function and the probabilistic constraint functions.

| Performance functions | Descriptions | Corresponding objective function or constraint functions |
|-----------------------|--------------|--------------------------------------------------------|
| \( h_0(k, \theta) \)  | Maximum negative deviation of gust reaction | \( \mathbb{P}[h_0(k, \theta) < -0.45] \) |
| \( h_1(k, \theta) \)  | Gain margin  | \( \mathbb{P}[h_1(k, \theta) < 6\text{dB}] < 10^{-6} \) |
| \( h_2(k, \theta) \)  | Phase margin | \( \mathbb{P}[h_2(k, \theta) < 45^\circ] < 10^{-6} \) |
| \( h_3(k, \theta) \)  | Overshoot of step response | \( \mathbb{P}[h_3(k, \theta) > 20\%] < 0.1 \) |
| \( h_4(k, \theta) \)  | 80% rising time of step response | \( \mathbb{P}[h_4(k, \theta) > 18] < 0.1 \) |
where $c$ is the simulation time, $C_{d,1}$ and $C_{d,2}$ are the thresholds for the mean square values. $c_1(k)$ and $c_2(k)$ limit the oscillation of control input $q_{cmd}$ and system output $n_z$.

The rare-event probabilities are estimated by the proposed SBSS (2000 samples at each level) while the others are approximated by MCS (10^5 samples) based on the surrogate model constructed by the 5th-order PCE. The expansion coefficients are computed employing the spectral projection with a tensor product quadrature (6 nodes in each dimension and 216 nodes in total). To guarantee the fulfillment of the rare-event requirements, the 3-$\sigma$ upper bounds are considered to satisfy the constraints:

\[
\begin{align*}
(1 + 3c_{v,1}) P[h_1(\theta) < 6dB] &< 10^{-6}, \\
(1 + 3c_{v,2}) P[h_2(\theta) < 45^\circ] &< 10^{-6},
\end{align*}
\]

where $c_{v,1}$ and $c_{v,2}$ are the c.o.v. of the estimates of $P[h_1(\theta) < 6dB]$ and $P[h_2(\theta) < 45^\circ]$ respectively. The c.o.v. can be reduced by increasing the number of samples at each level of SBSS. This, however, increases the computational burden. Therefore, the choice of the sample number is a tradeoff between the conservativeness of constraint bounds and the computational cost.

The optimization problem is solved by the Matlab global solver surrogateopt [29]. Given the designed control gains, the violation probabilities of performance functions $p_v$ and the numbers of calls to the true model $N_{call}$ are shown in Table 5. The results of MCS (10^4 samples) and SuS are regarded as references. The probabilistic bounds of gust reaction and step response are depicted in Fig. 12. As given in Table 5 and Fig. 12, the chance constraints in Table 4 are fulfilled directly during the minimization of the object function. Meanwhile, the results of surrogate-based methods match well with those of MCS and SuS, but far fewer true model evaluations are required in surrogate-based methods. This suggests that the surrogate-based methods are able to provide reasonably accurate estimates with much higher efficiency for both failure probability estimation and chance-constrained optimization.

Table 5 The violation probabilities of performance functions.

| Violation probabilities | MCS or SuS $p_v$ ($N_{call}$) | Surrogate-based methods $p_v$ ($N_{call}$) |
|-------------------------|-------------------------------|------------------------------------------|
| $P[h_0(k, \theta) < -0.45]$ | 9.53% (10000) | 9.47% (216) |
| $P[h_1(k, \theta) < 6dB]$ | $2.71 \times 10^{-7}$, $c_v \in [0.27, 0.52]$ (12800) | $2.04 \times 10^{-7}$, $c_v \in [0.26, 0.50]$ (1536) |
| $P[h_2(\theta) < 45^\circ]$ | $< 1 \times 10^{-10}$ (18200) | $< 1 \times 10^{-10}$ (2196) |
| $P[h_3(\theta) > 20\%]$ | 0.82% (10000) | 0.87% (216) |
| $P[h_4(\theta) > 1s]$ | 8.72% (10000) | 8.65% (216) |

VI. Conclusions

An RSM-based subset simulation method is developed and applied to estimate rare-event probabilities in the framework of performance-guaranteed control optimization with rare-event constraints. Compared with SuS, the proposed SBSS saves a large number of calls to the true model, thus gaining a high efficiency for both rare failure probability approximation and rare-event chance-constrained optimization. The simulation results demonstrate that SBSS provides a comparable performance to SuS but with much less computational expense and the performance-based control optimization directly ensures the fulfillment of requirements. We plan to extend this concept to solve high-dimensional problems in the future work.

Acknowledgments

The first author acknowledges the financial support from China Scholarship Council (CSC) on his doctoral study at TUM.
a) The probabilistic bounds of gust reaction  

b) The probabilistic bounds of step response

Fig. 12  The estimations of probabilistic bounds.

References

[1] Certification Specifications for All Weather Operations (CS-AWO), European Aviation Safety Agency, 2003.

[2] Löbl, D., and Holzapfel, F., “Subset Simulation for Estimating Small Failure Probabilities of an Aerial System Subject to Atmospheric Turbulences,” AIAA Atmospheric Flight Mechanics Conference, AIAA, 2015, pp. 1–11.

[3] Beaverstock, C., Maheri, A., Richardson, T., Lowenberg, M., and Isikveren, A., “Methods for Conceptual Flight Control System Design,” 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, AIAA, 2009, pp. 1–19.

[4] Au, S. K., and Beck, J. L., “Estimation of Small Failure Probabilities in High Dimensions by Subset Simulation,” Probabilistic Engineering Mechanics, Vol. 16, No. 4, 2001, pp. 263–277.

[5] Au, S. K., and Wang, Y., Engineering Risk Assessment with Subset Simulation, John Wiley & Sons, 2014.

[6] Li, B., and Der Kiureghian, A., “Accelerating Subset Simulation with a Surrogate Model,” IFIP Working Group 7.5 on Reliability and Optimization of Structural Systems, 2016, pp. 1–10.

[7] Dubourg, V., Sudret, B., and Bourinet, J. M., “Reliability-Based Design Optimization Using Kriging Surrogates and Subset Simulation,” Structural and Multidisciplinary Optimization, Vol. 44, No. 5, 2011, pp. 673–690.

[8] Bourinet, J. M., Deheeger, F., and Lemaire, M., “Assessing Small Failure Probabilities by Combined Subset Simulation and Support Vector Machines,” Structural Safety, Vol. 33, No. 6, 2011, pp. 343–353.

[9] Bourinet, J. M., “Rare-Event Probability Estimation with Adaptive Support Vector Regression Surrogates,” Reliability Engineering & System Safety, Vol. 150, 2016, pp. 210–221.

[10] Papadopoulos, V., Giovanis, D. G., Lagaros, N. D., and Papadrakakis, M., “Accelerated Subset Simulation with Neural Networks for Reliability Analysis,” Computer Methods in Applied Mechanics and Engineering, Vol. 223, 2012, pp. 70–80.

[11] Vitus, M. P., “Stochastic Control via Chance Constrained Optimization and its Application to Unmanned Aerial Vehicles,” Ph.D. Thesis, Stanford University, Stanford, USA, 2012.

[12] Zhao, Z., Liu, F., Kumar, M., and Rao, A. V., “A Novel Approach to Chance Constrained Optimal Control Problems,” 2015 American Control Conference, IEEE, 2015, pp. 5611–5616.

[13] Piprek, P., Gros, S., and Holzapfel, F., “Rare Event Chance-Constrained Optimal Control Using Polynomial Chaos and Subset Simulation,” Processes, Vol. 7, No. 4, 2019, p. 185.

[14] Mesbah, A., “Stochastic Model Predictive Control: An Overview and Perspectives for Future Research,” IEEE Control Systems Magazine, Vol. 36, No. 6, 2016, pp. 30–44.
