Solution of the Benchmark Control Problem by Scenario Optimization

Conference Paper · October 2019

DOI: 10.1115/DSCC2019-8949

Roberto Rocchetta
Security and Embedded Netowrked Ssytems W&I

Proceedings of the ASME 2019
Dynamic Systems and Control Conference
DSCC2019
October 8-11, 2019, Park City, UT, USA

DSCC2019-8949

RELIABILITY-BASED DESIGN BY SCENARIO OPTIMIZATION

Roberto Rocchetta[1]
National Institute of Aerospace (NIA),
Hampton, VA, USA.

Luis G. Crespo[2], **Sean P. Kenny**
Dynamic Systems and Control Branch, NASA
Langley Research Center, Hampton, VA, USA

ABSTRACT

This article introduces a scenario optimization framework for reliability-based design given measurements of the uncertain parameters. In contrast to traditional methods, scenario optimization makes direct use of the available data thereby eliminating the need for assuming a distribution class and estimating its hyper-parameters. Scenario theory provides formal bounds on the probabilistic performance of a design decision and certifies the system ability to comply with various requirements for future/unseen observations. This probabilistic certificate of correctness is non-asymptotic and distribution-free. Furthermore, chance-constrained optimization techniques are used to detect and eliminate the effects of outliers in the resulting optimal design. The proposed framework is exemplified on a benchmark robust control challenge problem having conflicting design objectives.

Keywords: Reliability-based design optimization, scenario optimization, probability of failure, outliers.

1. INTRODUCTION

Reliability-Based Design Optimization (RBDO) methods have been historically developed to identify reliable system designs and by explicitly modeling relevant sources of uncertainty. RBDO approaches have proven their strength in different fields and disciplines, to name a few, in off-shore energy systems optimization, [1], in composite structures design, [2], or in improving the performance of turbomachinery aerodynamics, [3].

To proceed with RBDO analysis, however, a well-suited model for the uncertainty should be provided and the performance of the final design will inevitably rely upon its goodness. This can be seen as a shortcoming of existing RBDO methods as, in many situations, a good uncertainty model can be a very complex task requiring multivariate density estimation and dependency modeling. As an example, think about defining a joint PDF fitted on the available data when the data are costly and thus limited or when the uncertain inputs are strongly dependent. In those situations, the analyst will be forced to introduce unwarranted model assumptions to describe the uncertainty, most likely affecting the final design.

Differently from distribution-based RBDO methods, data-driven optimization approaches do not rely on a probabilistic characterization of the uncertainty to seek and optimal design. In particular, Scenario optimization provides a means for data-driven decision-making, [4]. In recent years, Scenario theory has grown in popularity thanks to its high degree of generality, and its ability to provide a theoretical certificate of robustness for the optimal solution (i.e. the probability of the decision to suitable to yet unseen scenarios given a degree of confidence), see e.g. [5].

Scenario theory has been extensively studied for convex programs. In such cases, the degree of generalization of a decision (robustness) to yet unseen scenarios can be determined in advance. In [6] an upper bound on the violation probability provided and, similarly, [4] investigated a lower bound on the samples size needed to guarantee a robustness degree. The loss in robustness resulting from the removal of outliers in the dataset is studied in [7]. More recently, [8] presented the Wait-and judge method, which can be used to prescribe a-posteriori bounds on the robustness (i.e. only after the solution has been computed) of optimal designs found by both solving convex and non-convex optimization programs. Refinement of the Scenario theory to non-convex classes of problems has been investigated by different authors, see for instance, [5], [8], and [9].

[1] Contact author: roberto.rocchetta@nianet.org
[2] Contact author: luis.g.crespo@nasa.gov

In this paper, a RBDO framework is proposed to minimize the probability of violating conflicting requirements. The proposed method is compared against alternative data-driven approaches and the quality of the solution discussed on the basis of the probabilistic performance of the optimal design. Probabilistic guarantees on the optimal design robustness are obtained using non-convex scenario theory. The proposed method is tested on a modified version of the benchmark problem for robust control design proposed by [10].

The structure of this paper is as follows: Section 2 introduces the theoretical background on RBDO. Scenario theory is introduced in Section 3 and the proposed optimization framework is described In Section 4. Two scenario optimization schemes are tested on a modified version of a controls challenge problem introduced in Section 5. Results are discussed in Section 6, and Section 7 closes the paper with preliminary conclusions and future research directions.

2. RELIABILITY-BASED DESING-OPTIMIZATION

The goal of RBDO problems is to find a design d^* that minimizes the probability of failure P_f, i.e.,

$$\begin{aligned} d^* &= \arg\min_{\mathrm{d}} P_f\,(d) \\ P_f(d) &= \int_{F(\delta,\mathrm{d})} f_\delta d\delta \end{aligned} \tag{1}$$

where f_δ is the joint probability distribution function of uncertain parameters $\delta \in \Delta \subseteq \mathbb{R}^{n_g}$, d is the design variable $d \in \Theta \subseteq \mathbb{R}^{\mathrm{n_d}}$, and $F(\delta, \mathrm{d})$ is the composite failure domain defined as follows:

$$F(\delta,\mathrm{d}) = \cup_{i=1}^{n_g} F^{\mathrm{i}}(\delta,\mathrm{d}) \tag{2}$$

where $F^i(\delta,d) = \{\delta\colon g_i(\delta,d) \geq 0\}$ is the individual failure domain corresponding to the i^{th} requirement whereas the composite failure domain $F(\delta,\mathrm{d})$ corresponds to all n_g reliability requirements. $g \in \mathbb{R}^{n_g}$ is the system performance function. The complement of the failure domain is called the safe domain. A design d satisfies the requirements for parameter δ if $g_i(\delta\,,d) < 0, \sim \forall\, i \in \{1,..,n_g\}$.

Solving Eqn. (1) entails evaluating a multidimensional integral repeatedly. The high computational cost of estimating this integral has led to the use of asymptotic and sampling-based approximations e.g. FORM and SORM methods, Monte Carlo simulation (MCS), and other sampling-based approaches. Once an estimator $\hat{P}_f(d)$ is computed, an optimization algorithm can be readily applied. Note that the uncertainty model $f_\delta(\delta)$ is a key component of the process and, for instance in a MCS, it is necessary to sample m realizations of $f_\delta(\delta)$ to estimate:

$$\hat{P}_f(\delta,d) = \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}_{w(\delta,d)>0} \tag{3}$$

where $w(d,\delta) = max_j\, g_j(d,\delta)$ is the worst-case performance function, and $\mathbf{1}$ is the indicator function.

Providing a good probabilistic model of δ can be a challenging task in many situations due to lack of data, complex dependencies, and high dimensions thereby requiring unwarranted assumptions. These, often unsubstantiated, assumptions might inevitably affect the optimal design. The formulation below seeks an optimal design that minimizes the chance of violating the system requirements without any assumption on the distribution for δ.

3. SCENARIO OTIMIZATION

Scenario approaches have grown in popularity thanks to their data-driven setting, versatility and theoretical certificate of generalization which can be obtained for the proposed solution, [5]. Methods grounded on the Scenario theory have been used to address: robust optimization problems [4], [11], feasibility problems [7], regression and prediction problems, [12], [13].

To better understand the concepts on which Scenario theory is grounded, it is convenient to define mathematically a scenario optimization program. Consider a probability space Δ, equipped with a σ-algebra and a probability measure $\mathbb{P}$. A set of IID observations $\mathcal{D}_\delta = \{\delta^{(1)}, \delta^{(2)}, \dots, \delta^{(N)}\}$, e.g. samples of the uncertain parameters in (1), are drawn from Δ according to a stationary and unknown distribution $\mathbb{P}$. Each observation $\delta^{(i)} \in \Delta$ defines a so-called scenario and a Scenario Program SP, the decision maker seeks to find an optimal decision (i.e. an optimal design in this work) $d^* = \mathcal{SP}(\mathcal{D}_\delta)$ such that:

$$\mathcal{SP}(\mathcal{D}_\delta) := \begin{cases} \quad \arg\min_{d} f(d) \\ s.t.\;\; d \in \Theta_{\delta^{(i)}} \forall\, \delta^{(i)} \in\, \mathcal{D}_\delta \end{cases} \tag{4}$$

where d is the optimization variable, Θ is the decision space, $f\colon \Theta \mapsto \mathbb{R}$ is any function (e.g. a cost or unreliability function) and $\Theta_{\delta^{(i)}} \subseteq \Theta$ is the constrained decision space, i.e. a set of designs satisfying the system requirements for scenario $\delta^{(i)}$. With this level of generality, the only assumptions are $\mathbb{P}$ being stationary and the elements of $\mathcal{D}_\delta$ being IID.

3.1 Certify ε-robustness

Scenario theory can be used to assess how well d^* generalizes to yet unseen situations $\delta \in \Delta$, thus providing a powerful robustness-monitoring capability. For this purpose, it is useful to define a violation probability $V(d^*) = \mathbb{P}\{\delta \in \Delta\colon d^* \in \Theta_\delta\}$ and a reliability parameter $\epsilon \in (0,1)$. An ϵ-robust (or ϵ-feasible) design d^* is a solution for which $V(d^*) \leq \epsilon$, [14]. In other words, an ϵ-robust solution will comply with the requirements for new scenarios with probability 1-ϵ.

The violation probability, $V(d)$, is inherently stochastic, as it depends on the random set of scenarios $\mathcal{D}_\delta$. It has been proven that for convex and fully-supported problems the distribution of $V(d)$ is dominated by the Beta distribution, independently from the probability $\mathbb{P}$ generating $\mathcal{D}_\delta$. This result, together with a provided confidence parameter (β), can be used to compute a tight upper bound on the optimal d ϵ-robustness $\mathbb{P}^{\mathrm{N}}[V(d^*) \leq \epsilon] \leq 1-\beta$. This is valid for convex problems for which given $N > n_d$, where n_d is the number of design variables, and N the sample size, the cardinality of the set of support constraints3, s_N^*, is equal to the dimension of the design space n_d with probability 1, [6]. However, convex problems are often only partially supported and real-world design problems are often non-convex. Recently, [14] and extended scenario theory to non-convex and partially supported problems. Specifically, they show that

$$\mathbb{P}^{\mathrm{N}}[V(d^*) \leq \epsilon(s_N^*)] \leq 1-\beta \tag{5}$$

where ϵ is a function of the cardinality of the set of support constraints s_N^* (which is a computable quantity and sometimes reference to as solution complexity) given by

$$\epsilon(k) = \begin{cases} 1 & \text{if} \quad k = N \\ 1 - \left(\frac{\beta}{N\binom{N}{k}}\right)^{\frac{1}{N-k}} & \text{otherwise} \end{cases} \tag{6}$$

This result shows that a high complexity of a decision (s_N^*) corresponds to low generalization guarantees, i.e., lower robustness. Moreover, it shows that solutions with better coverage of Δ (i.e. higher N), but same complexity s_N^*, have higher robustness. Figure 1 shows two reliability curves, on the y-axis, versus N (x-axis). The two curves are obtained for $s_N^* = 2$ (dotted line), $s_N^* = 6$ (solid line) and a confidence $\beta = 1e\text{-}6$.

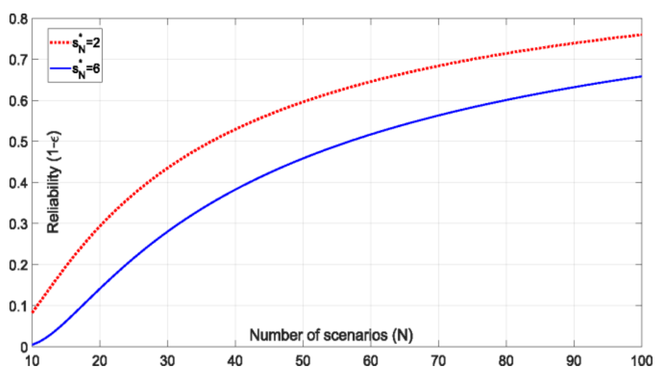

FIGURE 1: PLOT OF 1- ϵ , k=2 AND k=6 FOR INCREASING N AND β =1e-6.

Thus, to better assess the generalization properties of an optimal design, the analyst must find the support set with the smallest cardinality possible. However, for the purposes of applying Eqn. (5) minimal cardinality is not required (although this practice yields the tightest bound). The search for the set of support constraints is often done numerically, [14].

4. RBDO BY SCENARIO

A scenario RBDO formulation is:

$$\mathcal{SP}_1(\mathcal{D}_\delta) := \begin{cases} \arg \min\limits_{d,\gamma} \gamma \\ s.t.\ w\left(d, \delta^{(k)}\right) \leq \gamma \\ \delta^{(k)} \in \mathcal{C}_\delta \subseteq \mathcal{D}_\delta \end{cases} \tag{7}$$

where $\gamma \in \mathbb{R}$. Hence, $\mathcal{SP}_1$ seeks to minimize an upper bound to the worst-case performance function for all the elements in the data sequence. When the optimal cost is negative, the optimal design satisfies all the requirements for all observations. The freedom in choosing $\mathcal{C}_\delta$ enables solving different problems. For instance, the subset $\mathcal{C}_\delta$ can be the full set of observations $\mathcal{D}_\delta$. Alternatively, $\mathcal{C}_\delta$ can be used to eliminate the effects of outliers in the data sequence. Determining which scenario is an outlier is a process that might depend on the design variable itself. Using the worst-case performance function w defined earlier as a figure of merit, and denoting as F_w the empirical cumulative distribution of w corresponding to $\mathcal{D}_\delta$, the subsample

$$\mathcal{C}_\delta = \left\{\delta \in \mathcal{D}_\delta : F^{-1}_{w(\delta,\mathcal{D}_\delta)}(1-\alpha) \leq \gamma\right\} \tag{8}$$

only contains the $1-\alpha$ percent of the data with the best performance. Note that making $\alpha = 0$ makes $\mathcal{C}_\delta(d) = \mathcal{D}_\delta$. In this setting, the scenario program above can be written as:

$$\mathcal{SP}_1(\mathcal{D}_\delta, \alpha) := \arg \min_{d,\gamma} \{\gamma : F^{-1}_{w(d,\mathcal{D}_\delta)}(1-\alpha) \leq \gamma\} \tag{9}$$

Another program of interest is:

$$\mathcal{SP}_2(\mathcal{D}_\delta) := \arg \min_{d,\gamma} \left\{\gamma : \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{w(d,\delta^{(i)})>0} < \gamma\right\} \tag{10}$$

where $\gamma \in \mathbb{R}$. Hence, (10) seeks the design point that minimizes the empirical probability of failure. As such, this formulation will be denoted as $\mathcal{SP}_2(\mathcal{D}_\delta)$. Note that $\mathcal{SP}_2(\mathcal{D}_\delta)$ has a piecewise constant constraint making gradient-based algorithms inapplicable.

[3] A constraint of (3) is a support constraint if its removal changes the optimal design. The set of support constraints (or support set) $\mathcal{S} \subseteq \mathcal{D}_\delta$ is a k-tuple $\mathcal{S} = \{\delta^{(11)}, \ldots, \delta^{(ik)}\}$ which gives the same solution as the original sample, i.e. $\mathcal{SP}(\mathcal{S}) = \mathcal{SP}(\mathcal{D}_\delta)$ is a minimal support set if no further scenarios can be removed without changing the solution.

4.1 Identify Support Constraints

A standard procedure to identify the set of support constraints consists in removing one scenario at a time from $\mathcal{D}_\delta$ and checking whether the optimized (d^*, γ^*) obtained for $\mathcal{D}_\delta/\mathrm{i}$, where $/\mathrm{i}$ stands for a removal of the scenario i, is equal to the original solution. If $\mathcal{SP}(\mathcal{D}_\delta) = \mathcal{SP}(\mathcal{D}_\delta/\mathrm{i})$, then i is not a support scenario and removed from $\mathcal{D}_\delta$. The procedure when repeated converges to a set of support constraints. However, this procedure can be very time consuming (or even time infeasible), especially for large N and computationally intensive solvers $\mathcal{SP}$. The program $\mathcal{SP}(\mathcal{D}_\delta/\mathrm{i})$, will have run N times with $N-k$ constraints, where k is the number of removed scenarios. The number of constraints $N-k$ will be high for most of the procedure.

In this work, an efficient algorithm is proposed to identify s_N^* for the optimal scenario decision $\theta^* = (d^*, \gamma^*)$. A higher efficiency is achieved by adding worst-case-scenario one-at-a-time rather than removing scenarios. The procedure works as follows:

1) Solve $\theta^* = \mathcal{SP}(\mathcal{D}_\delta)$ and for initial guess θ_0;
2) Set the initial solution to an arbitrary initialization $\theta_{old} = (d_0, \gamma_0)$, e.g. close to the initial θ_0;
3) Initialize an empty set of support constraint $\mathcal{S} \leftarrow \emptyset$
4) For each $\delta \in \mathcal{D}_\delta$ and in correspondence of θ_{old} compute $w(d, \delta)$;
5) Find the worst-case scenario constraint $i = \arg\max 1_i \approx w\left(d, \delta^{(i)}\right)$, update the support set $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$ and the scenario set $\mathcal{D}_\delta \leftarrow \mathcal{D}_\delta / i$
6) Compute optimal solution $\theta_{new} = SP(\mathcal{S})$ for the initial (d_0, γ_0)
7) If $\frac{||\theta_{new} - \theta^*||}{|\theta new|} \leq r_\theta$ stop procedure and return the support set $\mathcal{S}$ and its cardinality, $s_N^* = |\mathcal{S}|$; Otherwise, set $\theta_{old} = \theta_{new}$ and go to step (4);

The procedure converges when the relative errors between θ_{new} and the initial θ^* is less than a tolerance r_θ. It returns a (possibly reducible) support set, i.e. set for which $SP(\mathcal{D}_\delta) = SP(\mathcal{S})$. An irreducible set of minimum cardinality can be obtained by removing elements from $\mathcal{S}$, such that $SP(\mathcal{D}_\delta) = SP(\mathcal{S}/i)$. A stopping criterion θ_{new} is introduced in the step (7) of the algorithm (e.g. r_θ=1e-6) to avoid the procedure from getting stuck due to numerical approximations and local plateau in the (d, γ) space.

5. CASE STUDY: ROBUST CONTROL CHALLENGE

A modified version of the benchmark problem on robust control design, see [10], is used to test the proposed RBDO method. The system is the two-mass spring system shown in Fig. 2.

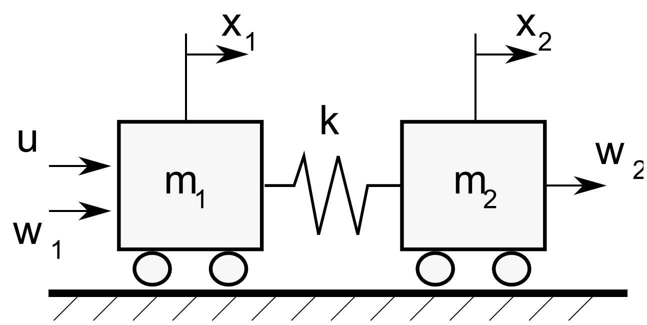

FIGURE 2: THE TWO-MASS SPRING SYSTEM, BENCHMARK FOR ROBUST CONTROL DESING.

A control force acts on the first mass, and the position of body 2 is measured. In the modified version of the system we further consider a non-linear spring constant k_n a time delay τ (i.e. a first order lag between controller signal u_c and actuator response u) and an uncertainty factor λ on the loop-gain, due to multiplicative variation in observation, control gain and/or actuator failure. The resulting state-space equations governing the system dynamics are:

$$\ddot{x}_1 = \frac{k_l}{m_1}(x_2 - x_1) + \frac{k_n}{m_1}(x_2 - x_1)^3 + \frac{\lambda u}{m_1}$$
$$\ddot{x}_2 = \frac{k_l}{m_2}(x_1 - x_2) + \frac{k_n}{m_2}(x_1 - x_2)^3 + \frac{w_2}{m_2}$$
$$\tau \dot{u} = u_c - u$$

where $x_1(t)$ is the position of the first mass, $x_2(t)$ is the position of the second mass, $u(t)$ is the actuator control input, k_l is the linear coefficient of the spring, m_1 mass of the first body, m_2 mass of the second body, w_2 is a disturbance on the second body and $\lambda u = (u + w_1)$ combines the actuator response and the disturbance on the first body, w_1. The goal of the problem is to design a linear feedback compensator such that the following reliability requirements are satisfied:

1) Local closed-loop stability;
2) Setting time: position of the mass 2 must fall between ± 0.1 after 15 seconds;
3) Control effort, i.e. the control signal must fall between ± 1;

These requirements lead to the performance functions $g(d, \delta) = [g_1, g_2, g_3] \in \mathbb{R}^3$:

$$g(d, \delta) = \begin{bmatrix} \max_{1 \leq i \leq n_\delta} \mathcal{R}\left[s^{(i)}\right] \\ \max_{t \geq 15} |x_2(t)| - 0.1 \\ \max_{t > 0} |u(t)| - 1 \end{bmatrix}$$

where $s^{(i)}$ is the closed loop pole of the linearized system, and $\mathcal{R}[\,]$ is real part operator. Note that the time domain requirements require simulating the time response of the system by numerical integration. The state-space representation of the controller is given by:

$$\dot{x_c} = Ax_c + By$$
$$u_c = Cx_c + Dy \tag{11}$$

where x_c is the controller state, and A, B, C, D are the controller matrices. The canonical form in Eqn. (11) can be conveniently rewritten in a single input single-input transfer function $H(s) = C(sI - A)^{-1}B + D$, where H reads:

$$H(s) = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$$

The controller design is thus defined by the coefficient of the single-input transfer function $d = [b_0, b_1, b_2, b_3, a_0, a_1, a_2, a_3, a_4] \in \mathbb{R}^9$. To enable the visualization of results only the uncertainty in the masses is considered. All other parameters assume the constant values prescribed in [10]. $N = 1000$ observations are available and Fig. 3 shows the elements in the data sequence $\mathcal{D}_\delta$, $\delta^{(i)} = [m_1^{(i)}, m_2^{(i)}] i = 1, \dots N$.

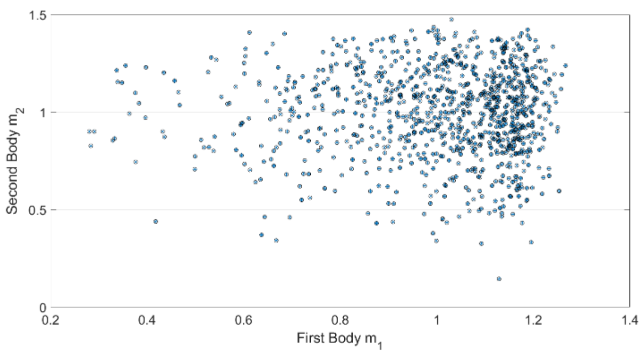

FIGURE 3: THE 1000 OBSERVATIONS COLLECTED FOR THE 2 MASSES.

It is worth emphasizing that thanks to the nonlinear dependency of the requirement function on the controller gains an a-priori probabilistic statement on the generalization guarantees cannot be made. For this problem, a-posterior assessment of the ϵ-robustness will be considered for an optimized design. This will be done by a-posteriori enumerating the support constraints for a solution d^*.

6. RESULTS AND DISCUSSION

The system reliability performance is computed for a nominal baseline design d_n, see [10]. The corresponding failure probability is $\hat{P}_f = 0.734$ (734 out of 1000 scenarios failed), whereas the maximum value of the performance functions $\overline{g}_j(d) = \max_i g_j\left(d, \delta^{(i)}\right)$ leads to $\overline{g}(d_n) = [0.0007, 1.974, 0.202]$. The worst scenario for d_n is $(m_1, m_2) = (1.1287, 0.1450)$ kg, for which the second body will have a maximum displacement of $x_2(t > 15) = \pm 2.074$, thus exceeding of 1.974 the threshold requirement of ± 0.1. Figure 4 shows the individual and overall failure and safe domains for the design d_n. The size of the safe domain is the smallest for requirement 2.

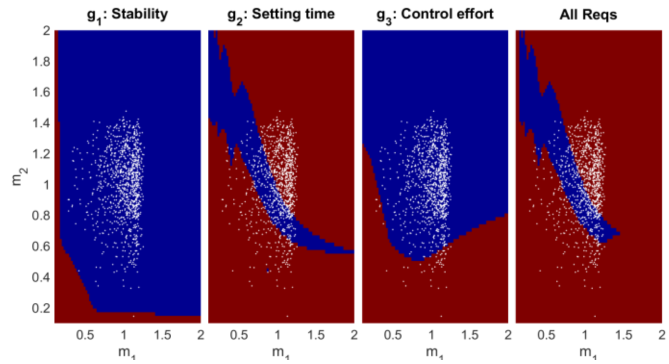

FIGURE 4: ON THE RIGHT PANEL, THE FAILURE DOMAIN (REGION IN RED) FOR THE BASELINE DESING AND THE SCENARIOS (WHITE DOTS). THE 3 PANELS ON THE LEFT SHOW THE INDIVIDUAL $F^i(d_n)$ FOR THE 3 REQUIREMENTS.

TABLE 1: FAILURE PROBBAILITY, WORST-CASE PERFORMANCE AND OPTIMAL DESING.

	$\mathcal{SP}_1(\mathcal{D}_\delta, 0)$	$\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$	$\mathcal{SP}_2(\mathcal{D}_\delta)$	Nominal
$\hat{P}_f$	0.981	0.057	0.175	0.734
$\overline{g}_1$	-0.0272	-0.013	0.069	$7e-4$
$\overline{g}_2$	0.5925	1.576	3.054	1.974
$\overline{g}_3$	0.5925	0.193	1.406	0.202
a_4	0.2238	0.5375	0.7600	0.5503
a_3	0.6811	1.3346	1.9491	1.4175
a_2	3.1275	2.4206	3.0497	2.6531
a_1	2.3615	2.1689	2.7344	2.4802
a_0	1.1833	0.8084	1.0594	1.0000
b_3	-0.0982	2.4802	-0.0831	0.1324
b_2	0.4702	0.6146	0.6358	0.3533
b_1	0.5886	0.5265	0.7752	0.6005
b_0	0.0777	0.0716	0.0981	0.0728

The design approaches introduced above were used to seek optimal controllers. To this end, the design constraints $d_n - 0.8|d_n| \leq d \leq d_n + 0.8|d_n|$ are imposed. Two implementations of $\mathcal{SP}_1$ were considered: the first one uses the full data set whereas the second one eliminates $\alpha = 0.05$ percent of the data. Table 1 presents figures of merit and the optimal designs for the 3 approaches. $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ and $\mathcal{SP}_2(\mathcal{D}_\delta)$ reduced considerably the failure probability compared to the nominal design. However, the $\mathcal{SP}_2(\mathcal{D}_\delta)$ design worsens the less reliable scenario (e.g. $\overline{g}_2 = 3.054$). Conversely, $\mathcal{SP}_1(\mathcal{D}_\delta, 0)$ greatly reduced the consequences for the worst-case-scenario (e.g. $\overline{g}_2 = 0.5925$), nonetheless, it failed to comply with the reliability requirements for most of the scenarios, thus attaining a large failure probability ($\overline{g}_2 = 0.981$).

The 4 panels in Fig. 5 show the overall fail and safe region for the 4 designs, the nominal design (top-left panel), the design $\mathcal{SP}_1(\mathcal{D}_\delta, 0)$ (top-right), the design $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ (bottom-left) and $\mathcal{SP}_2(\mathcal{D}_\delta)$ (bottom-right). The 4 panels in Fig. 6 show the values of $w(d, \delta)$ against the scenario pairs (m_1, m_2). In Fig. 6 it can be seen that $\mathcal{SP}_1(\mathcal{D}_\delta, 0)$, top panel on the right, lowered the worst-case $w(d, \delta)$ values. However, in doing so, it kept most scenarios in the failure region (red circle markers). Conversely, $\mathcal{SP}_2(\mathcal{D}_\delta)$, in the bottom-right panel, significantly reduced number of failed scenarios but worsening the extreme cases (not the different scale on the z-axes). Similarly, $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ greatly improved the reliability of the system achieving a good compromise between worst-case mitigation, $\overline{g} = [-0.013, 1.576, 0.193]$, and minimization of the probability of failure, $\hat{P}_f = 0.057$.

Figure 7 presents the failure domains corresponding to $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$. Compared to the nominal design (Fig. 4), a small contraction of the safe domain for requirement 3 can be observed. On the other hand, the size of the failure region defined by requirement 2 (which was acting as a bottleneck for the nominal design) was greatly reduced, thus providing a substantial gain in terms of reliability (only 57 scenarios could not comply with the reliability requirement against the 734 of the nominal design).

6.1 Robustness Analysis

The ϵ-robustness of the scenario decision is examined by computing the number of support constraints s^*_{1000}. Support constraints for the the program $\mathcal{SP}_1(\mathcal{D}_\delta, 0)$ are identified using the support scenario identification procedure. A support set of size s^*_{1000}=11 is obtained and further reduced to s^*_{1000}=4 by removing scenarios from $\mathcal{S}$. Equation (6) is used to obtain the ϵ-robustness of d^*: $\mathbb{P}\{\delta \in \Delta: w(\delta, d^*) > 0.5925\} \leq \epsilon(s^*_{1000})$, where $\gamma^* = 0.5925$ and $\epsilon(4) = 0.0443$ for a confidence parameter $\beta = 1e-6$. In other words, a certificate of robustness for the optimal design $(d^*, \gamma^*) = \mathcal{SP}_1(\mathcal{D}_\delta, 0)$ has been provided. This ensures that, for 95.57 % of the unseen scenarios pairs (m_1, m_2) and with high confidence ($\beta = 1e-6$), the 3 performances function will result $g_j \leq 0.5925$.

The 4 support constraints are $\delta^{(1k)} = (1.128, 0.145)$, $\delta^{(2k)} = (0.42, 0.44)$, $\delta^{(3k)} = (1.25, 0.90)$ and $\delta^{(4k)} = (0.67, 0.34)$. Each panel in Fig. 8 displays on the y-axis the values for $w(\delta^{(ik)}, d^*)$, where $d^* = \mathcal{SP}_1(\mathcal{D}_\delta, 0)$ and for the 4 support constraints (solid lines). Each of the elements of d^* is explored in the interval $[d_n - 0.8|d_n|, d_n + 0.8|d_n|]$ as displayed on the x-axes. As expected, the d^* values (vertical dashed lines) are the one corresponding to a global minimization of w. It can be observed: (1) the non-convex behavior of the constraints (e.g. see the panel for b_1 on the top right-hand side); (2) Competitive scenarios preventing from further reducing w (e.g. $\delta^{(2k)}$ and $\delta^{(3k)}$ in the b_2 panel).

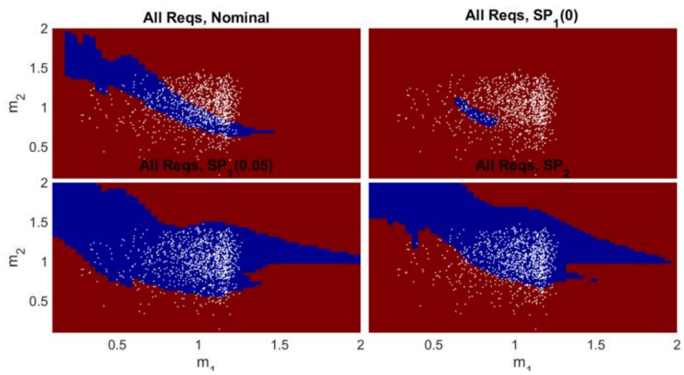

FIGURE 5: THE FAILURE AND SAFE DOMAINS FOR THE 4 DESIGNS.

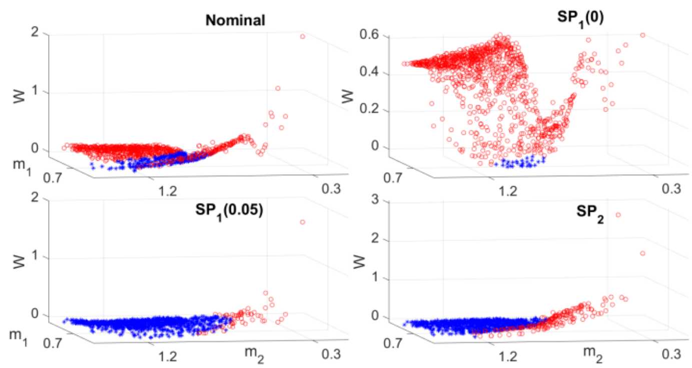

FIGURE 6: THE VALUES OF $w(d, \delta)$ (Z-AXIS) FOR THE 4 DESIGNS AND THE 1000 PAIRS (m_1, m_2).

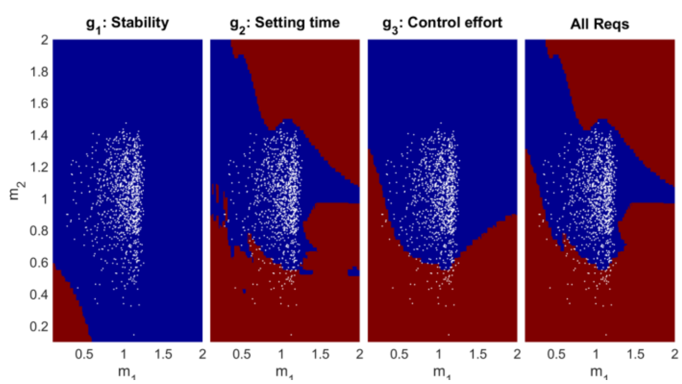

FIGURE 7: THE FAILURE DOMAIN FOR THE $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$.

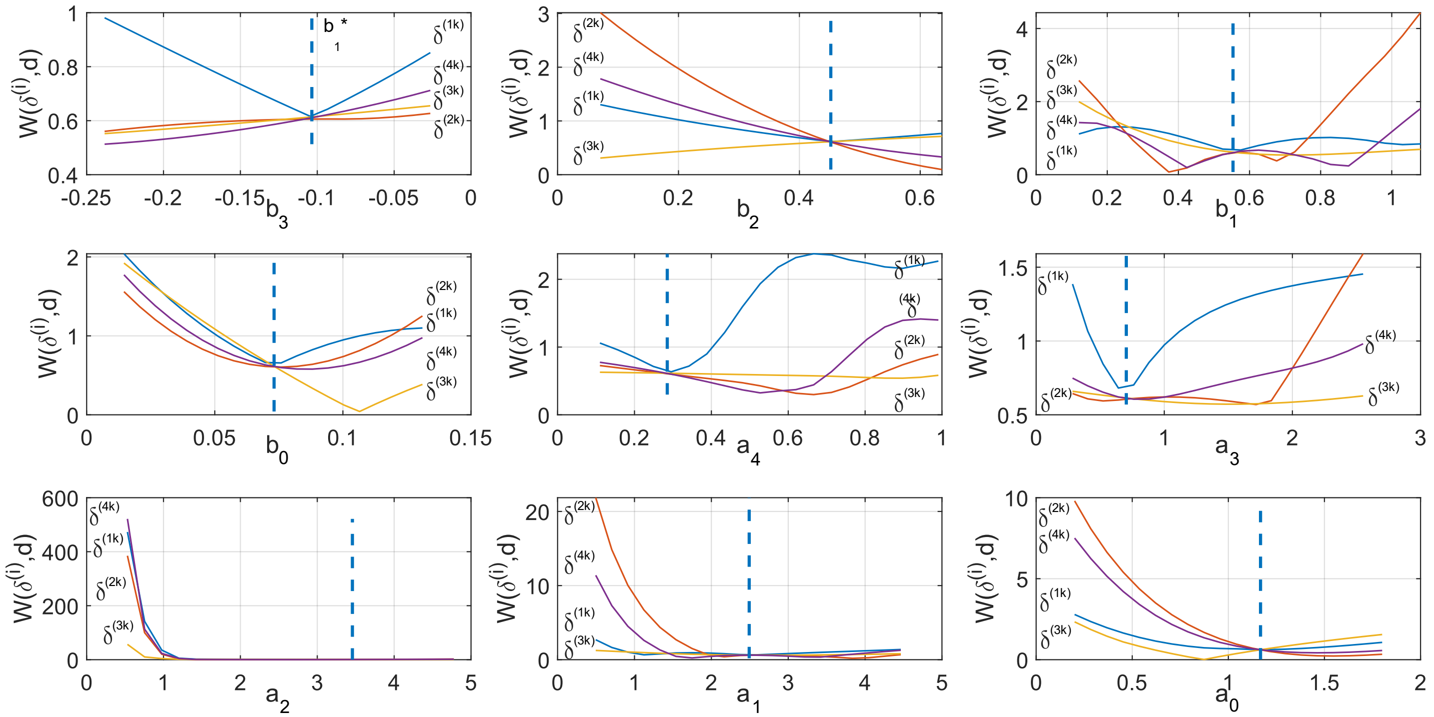

FIGURE 8: A VIEW ON $w(\delta^{(ik)}, d^*)$ WITH $d^* = \mathcal{SP}_1(\mathcal{D}_\delta, 0)$, $\delta^{(ik)} \in \mathcal{S}$ AND FOR DIFFERENT VALUE OF THE d^* ELEMENTS (THE 9 PANELS). THE 4 SUPPORT CONSTRAINTS ARE DISPLAYED IN SOLID LINES WILST d^* VALUES ARE PRESENTED BY DASHED LINES.

The problem $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ is somehow similar to the sample and discard approach presented in [7], but in a non-convex setting. The non-convex scenario theory is still in its infancy, and at this early stage a sound theoretical base to assess $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ robustness is not available. Nonetheless, it is intuitively arguable that the number of support constraints for $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$will be larger than the one of $\mathcal{SP}_1(\mathcal{D}_\delta, 0)$, and likely influenced by the choice on α. In that is the case, the robustness of $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ will be lower. The analysis of the robustness of $\mathcal{SP}_1(\mathcal{D}_\delta, 0.05)$ can be seen as a future goal once a theoretical development is provided.

Concerning $\mathcal{SP}_2$, the optimized design is expected to have a high number of the support constraints, possibly close (or even equal) to N. Further verification is required, however, this is potentially due to the high-complexity (in the scenario support sense) of a decision connected to a mean operator (to compute the probability of failure in the constraint). In other words, removal of any of the scenarios will affect the estimate $\frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{w(d,\delta^{(i)})>0}$ and, consequently, the optimal design parameter γ^*.

7. CONCLUSION

In this paper, a framework for reliability-based-design optimization based on Scenario theory has been investigated. By not requiring an explicit prescription of the uncertainty, the resulting optimal designs are exempt from the modeling error caused by such a practice. Several scenario-based formulations making direct use of available data are proposed and their corresponding robustness properties are assessed using scenario theory. Furthermore, a chance-constrained optimization formulation is proposed for eliminating the effects that outliers in the data set have in the resulting optimal design.

REFERENCES

[1] C. E. Clark and B. DuPont, “Reliability-based design optimization in offshore renewable energy systems,” *Renewable and Sustainable Energy Reviews*, vol. 97, pp. 390-400, 2018.

[2] S. Nikbakt, S. Kamarian and M. Shakeri, “A review on optimization of composite structures Part I: Laminated composites,” *Composite Structures*, vol. 195, pp. 158-185, 2018.

[3] Z. Li and X. Zheng, “Review of design optimization methods for turbomachinery aerodynamics,” *Progress in Aerospace Sciences*, vol. 93, pp. 1-23, 2017.

[4] G. C. Calafiore and M. C. Campi, “The scenario approach to robust control design,” *IEEE Transactions on Automatic Control*, vol. 51, pp. 742-753, 5 2006.

[5] M. C. Campi, S. Garatti and F. A. Ramponi, “A General Scenario Theory for Nonconvex Optimization and Decision Making,” *IEEE Transactions on Automatic Control*, vol. 63, pp. 4067-4078, 12 2018.

[6] M. Campi and S. Garatti, “The Exact Feasibility of Randomized Solutions of Uncertain Convex Programs,” *SIAM Journal on Optimization,* vol. 19, pp. 1211-1230, 2008.

[7] M. C. Campi and S. Garatti, “A Sampling-and-Discarding Approach to Chance-Constrained Optimization: Feasibility and Optimality,” *Journal of Optimization Theory and Applications,* vol. 148, pp. 257-280, 01 2 2011.

[8] M. C. Campi, S. Garatti and F. A. Ramponi, “Non-convex scenario optimization with application to system identification,” in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015.

[9] S. Grammatico, X. Zhang, K. Margellos, P. Goulart and J. Lygeros, “A Scenario Approach for Non-Convex Control Design,” *IEEE Transactions on Automatic Control,* vol. 61, pp. 334-345, 2 2016.

[10] B. Wie and D. S. Bernstein, “Benchmark Problems for Robust Control Design,” in *1991 American Control Conference*, 1991.

[11] M. M. Siraj, M. B. Saltik, P. M. J. V. Hof and S. Grammatico, “Scenario-based robust optimization of water flooding in oil reservoirs enjoys probabilistic guarantees,” *IFAC-PapersOnLine,* vol. 51, pp. 102-107, 2018.

[12] M. J. Lacerda and L. G. Crespo, “Interval predictor models for data with measurement uncertainty,” in *2017 American Control Conference (ACC)*, 2017.

[13] J. Sadeghi, M. Angelis and E. Patelli, “Frequentist history matching with Interval Predictor Models,” *Applied Mathematical Modelling,* vol. 61, pp. 29-48, 2018.

[14] M. C. Campi and S. Garatti, “Wait-and-judge scenario optimization,” *Mathematical Programming,* vol. 167, pp. 155-189, 01 1 2018.

[15] G. Calafiore and M. C. Campi, “Uncertain convex programs: randomized solutions and confidence levels,” *Mathematical Programming,* vol. 102, pp. 25-46, 01 1 2005.