AIAA JOURNAL
Vol. 59, No. 5, May 2021

Optimal Aircraft Design Decisions Under Uncertainty Using Robust Signomial Programming

Berk Öztürk* and Ali Saab†
Massachusetts Institute of Technology, Cambridge, MA 02139

https://doi.org/10.2514/1.J058724

Aircraft design benefits from optimization under uncertainty since design feasibility and performance can have large sensitivities to uncertain parameters. Legacy methods of uncertainty protection do not adequately explain the tradeoffs between feasibility and optimality, and they require prior engineering knowledge that may not be available for new system concepts. Furthermore, stochastic optimization over parameter distributions is computationally intractable for solving high-dimensional nonlinear design optimization problems. This paper proposes an efficient solution method for engineering design optimization problems under uncertainty using robust signomial programs (RSPs). Signomial programs (SPs) have demonstrated potential in solving multidisciplinary optimization problems, and the formulation of RSPs enables conceptual design that captures parametric uncertainty with probabilistic guarantees of constraint satisfaction. The proposed method transforms stochastic optimization problems to deterministic problems by considering the worst-case robust counterpart of each design constraint over a parameter uncertainty set, provided that each constraint is SP representable. The RSP formulation extends an existing robust geometric program (RGP) formulation by allowing difference-of-log-convex constraints that appear in many design problems. The RSP is solved efficiently and deterministically using a sequence of local RGP approximations. RSPs are then applied to unmanned aircraft design, and they are used to rigorously explore the tradeoff between robustness and optimality in design decisions.

I. Introduction

AEROSPACE design exists in a niche of design problems where "failure is not an option."‡ This is remarkable since aerospace design problems are rife with uncertainty about technological capabilities, environmental factors, manufacturing quality, and the future state of markets and regulatory agencies. Optimization under uncertainty seeks to provide designs that are robust to realizations of uncertainty in the real world and can reduce the high risk of aerospace programs.

Optimization has become ubiquitous in the design of engineered systems, and especially aerospace systems, as computing has improved dramatically and designs have continued to approach the limits of the second law of thermodynamics. Optimization under uncertainty has been identified by academia and industry as an area of opportunity in multiple review papers [1,2], and we detail some of its potential benefits as follows.

The uptake of new design tools in the aerospace industry has been low due to heavy reliance on legacy design methods and prior experience when faced with risky design propositions. Legacy tools have been predominant even in the design of novel configurations where experience in and understanding of the design tradespaces is lacking. Since new tools for design under uncertainty will better evaluate risk than legacy tools, there will be increased confidence in and uptake of new design tools.

Design under uncertainty will allow for a better understanding of the tradeoff between risk and performance. Optimization tools that rigorously consider uncertainty will yield designs that are less conservative than traditional designs while meeting the same reliability requirements. These tools will also better evaluate the viability of new concepts and configurations relative to legacy methods since they will capture the effects of technological uncertainty.

Finally, design under uncertainty will enable guarantees of constraint satisfaction under uncertainty. Designs will be more robust to uncertainties in manufacturing quality, environmental factors, technology level, and markets; and they will be better able to handle off-nominal operating conditions.

In economics, the idea that risk is related to profit is well understood and leveraged. In aerospace engineering, however, we often forget that risk aversity necessarily results in lower performance. Considering that conceptual design hedges against program risk, the tractable robust optimization (RO) frameworks proposed in this paper will give aerospace engineers the ability to rigorously tradeoff robustness to uncertainty with the performance penalties that result.

A. Approaches to Optimization Under Uncertainty

Faced with the challenge of finding designs that can handle uncertainty, the aerospace field has developed a number of methods to design under uncertainty. Oftentimes, aerospace engineers will implement *margins* in the design process to account for uncertainties in parameters that a design's feasibility may be sensitive to, such as material properties or maximum lift coefficient. Another traditional method of adding robustness is through *multimission design* [3], which ensures that the design is able to handle multiple kinds of missions in the presence of no uncertainty. This is a type of *finitely adaptive* optimization geared to ensure performance in off-nominal operations.

These legacy methods have several weaknesses. They provide no quantitative measures of robustness or reliability [1]. They rely on the expertise of an experienced engineer to guide the design process, without explicit knowledge of the tradeoff between robustness and optimality [2]. This is a dangerous proposition, especially in the conceptual design phase of new configurations, since prior information and expertise are not available. In these scenarios, it is especially important to implement physics-based tools to explore the design space [3]. Furthermore, legacy methods are often too conservative, ruling out potentially beneficial technologies and configurations due to the inability to adequately trade off performance and risk.

There are two rigorous approaches to solving design optimization problems under uncertainty, which are stochastic optimization (SO) and RO, contrasted in Fig. 1 and defined as follows. Note that stochastic optimization is an overloaded term, and it exists in at least two contexts in the literature. The first is the solution of deterministic problems with stochastic search space exploration. The second is the

Presented as Paper 2019-3351 at the AIAA Aviation 2019 Forum, Dallas, TX, June 17–21, 2019; received 11 June 2019; revision received 20 March 2020; accepted for publication 26 October 2020; published online 26 March 2021.

*Ph.D. Candidate, Department of Aeronautics and Astronautics.
†Master of Science 2018, Department of Aeronautics and Astronautics.
‡Quote is from Gene Kranz, The Mission Director of Apollo 13.

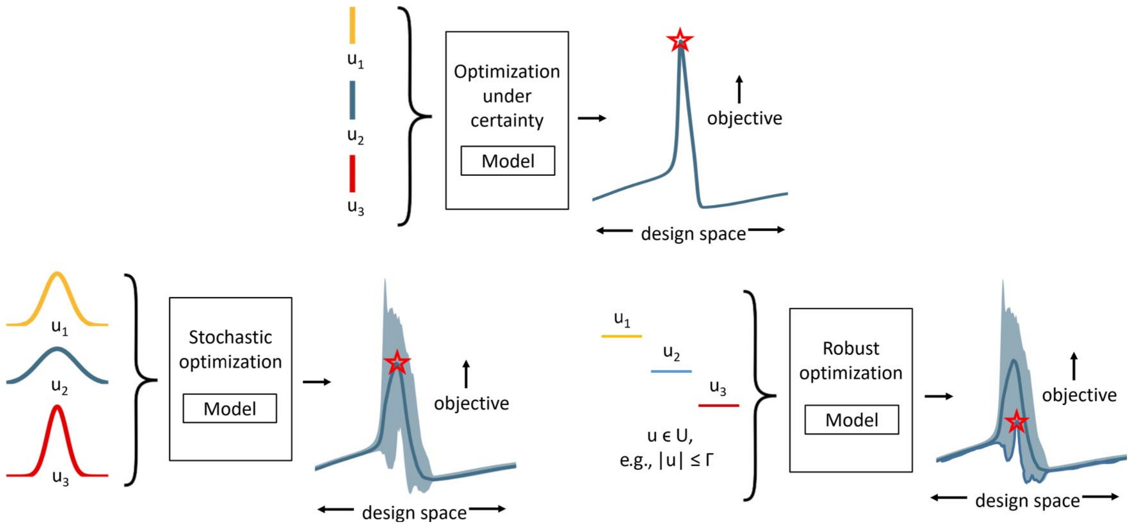


Fig. 1 SO and RO methods for optimization under uncertainty that use different definitions of uncertain inputs and produce different objective outcomes.

solution of design optimization problems with stochastic parameters, which is the focus of this paper.

SO pairs well with gradient-based approaches to solving nonlinear optimization problems such as those defined in Refs. [4–6]. These approaches implement an iterative process where the objective function and constraints are evaluated over an initial design, and first- and/or second-order information is used to converge the design toward a local optimum. In this context, SO problems deal with uncertainty by including probability distributions of the uncertain parameters in the iteration, as well as propagating the distributions through the physics of a design problem to ensure constraint feasibility with certain probability. The predominant goal of SO is to optimize some distributional characteristics (e.g., the mean as in Fig. 1) of the probability density function of the objective [7].

There have been recent developments in multimission aircraft design using SO. Liem et al. [5] propose the use of optimally weighted objective functions over an aircraft's operational design envelope for robust aircraft design. In following work, Liem et al. generate probability distributions of uncertain parameters from data and minimize the expectation of an objective function over parameter distributions [6]. Although these stochastic methods demonstrate significant improvements over legacy design methods in terms of design robustness, they do not address many of the aforementioned challenges of legacy design methods in capturing the robustness-optimality tradeoff. The scope of the design problems is narrow and limited to aerostructural optimization, and the number of uncertain parameters is low. The formulations assume the presence of data, limiting the effectiveness of the methods in conceptual design. They have large computational costs that are somewhat mitigated through surrogate modeling but would be detrimental in the conceptual design phase. Most importantly, they lack rigorous mathematical assessments of design feasibility under uncertain parameters.

In contrast to SO, RO can only be applied to mathematical programs that have a robust counterpart, such as linear, quadratic, semidefinite, and geometric programs. RO takes a different approach than SO in both the form of uncertain inputs and the objective functions. RO produces designs that are immune to constraint violations as long as parameter values come from within a defined uncertainty set. The objective of RO is to optimize the worst-case objective outcome of a design for a given set over the uncertain parameters. As such, RO avoids the need to sample and propagate probability distributions, and it turns SO problems into deterministic problems that are efficiently solved.

B. Comparison of Robust and Stochastic Optimization Methods for Conceptual Design

Both RO and SO have relative advantages in implementation. This paper will argue specifically that the formulation of conceptual engineering design problems under uncertainty as RO problems has advantages over SO formulations (a more mathematical programming-centric comparison is made in Ref. [8]).

1. Generality and Tractability

In the context of engineering, we claim that an optimization method is general when it can be used to solve a range of problems of interest. On the other hand, tractability describes whether or not the problems are solved to a satisfactory optimum within a reasonable computational time. Optimization under uncertainty is a difficult task that puts these two desirable subjective traits at odds with each other.

SO has the advantage of generality. SO methods are easily applicable to black box models or input–output systems. They require little knowledge, if any, about the constraints in the system of interest. RO methods are less general since they require the design objective and constraints to be explicit and cast in a form that has a worst-case counterpart. Thus, models for RO have to be transparent; and RO cannot be applied to black box models without significant prior data manipulation and fitting at a minimum. A mitigating factor is that many classes of conceptual engineering design problems can be cast or approximated in a form that is compatible with robust optimization.

On the other hand, RO is more tractable than SO due to the difference in method of uncertainty propagation. As mentioned in Sec. I.B, SO methods involve the propagation of probability densities throughout a model to determine their effects on constraint feasibility and the objective function. This requires the integration of the product of probability distributions with potential outcomes; and since the integration of continuous functions is difficult, this is often achieved through a combination of high-dimensional quadrature and discretizations of the uncertainty into possible scenarios. This propagation method results in a combinatorial explosion of possible outcomes that need to be evaluated to determine constraint satisfaction and the distribution of the objective. As a result, few problems can be addressed purely through SO (e.g., recourse problems [9,10]; the energy planning problem [11]; and certain aircraft design problems [5,6]), and even these are limited by combinatorics and costly system evaluations. Furthermore, they require problem-specific approximations so that generality is compromised. Robust versions of tractable optimization problems are not guaranteed to be tractable; but in practice, the aforementioned classes of optimization problems have tractable robust formulations [8]. In RO, there are no separate optimization and evaluation loops by construction, and thus RO problems can be solved to optimality many orders of magnitude faster than SO problems of the same form [8].

Conceptual design optimization values both generality and tractability: the former because engineers would like to apply methods for optimization under uncertainty without significant mathematical groundwork, and the latter because fast solution times are critical to reduce program risk early on in the design process when more aspects of the design are fluid. From this perspective, the relative intractability of SO-based approaches makes them unreliable for conceptual design since significant time is needed both to develop problem-specific tractable formulations and to find satisfactory optima. Furthermore, many engineering design problems such as aircraft design can be posed as optimization problems that have tractable robust counterparts, making RO better suited to conceptual design.

2. *Use of Data*

SO problems generally require complete knowledge of the probability distribution of parameters. RO requires only "modest assumptions about distributions, such as a known mean and bounded support" [12]. Since RO does not require as much information about uncertain parameters as SO does, it can better address conceptual design problems where there is a lack of experience or sparse and noisy data [8]. It is arguable that RO leaves a lot on the table by not taking advantage of distributional information about the uncertain parameters; however, there is a growing body of research on distributionally robust optimization [13,14], which seeks to leverage existing data.

3. *Stochasticity and Probabilistic Guarantees*

Although RO problems solve problems with uncertainty, RO formulations result in *deterministic*[§] solutions that are immune to all possible realizations of parameters in an uncertainty set [8]. There is extensive literature on RO methods that offer differing levels of conservativeness [15], depending on the kind of uncertainty set considered, that are guaranteed to be feasible over the uncertainty set of interest.

SO formulations provide no probabilistic guarantees since the optimum depends on realizations of random variables [16]. This is not satisfactory from an engineering perspective, since optimization runs over the same parameters may result in different solutions. Furthermore, designs can be sensitive to issues in sampling schemes over potentially unknown probability distributions. In the context of engineering design, the determinism and probabilistic guarantees of RO make it superior to SO.

It is important to highlight that, although both RO and SO seek to address the problem of optimization under uncertainty, they solve fundamentally different problems. In an ideal world where we have a problem that is tractable and globally optimal for both methods, the two different approaches would result in different solutions.

C. Geometric and Signomial Programming for Engineering Design

Geometric programming[¶] is a method of log-convex optimization that has been developed to solve problems in engineering design [17]. Although theory of the geometric program (GP) has existed since the 1960s, GPs have recently experienced a resurgence due to the advent of polynomial-time interior point methods [18] and improvements in computing. They have been applied to a range of engineering design problems with success. For a nonexhaustive list of examples, please refer to Ref. [19].

GPs have been effective in aircraft conceptual design [20,21]. However, the stringent mathematical form of a GP means it can only be applied to log-convex problems. The signomial program (SP) is the difference-of-log-convex extension of the GP that can be applied to solve this larger set of problems, albeit with the loss of some mathematical guarantees compared to the GP [22]. Aircraft pose some of the most challenging design problems [3], and signomial programming has been used to great effect in modeling and designing complex aircraft at a conceptual level quickly and reliably, as in Refs. [3,22,23]. Other interesting applications for SPs such as in network flow problems are being investigated.

[§]Determinism in this case refers to the outcomes of free variables in the optimization model. Different instances of a deterministic design problem with the same parameters will result in the same solution.

[¶]Programming refers to the mathematical formulation of an optimization problem.

Robust formulations exist for solving geometric programs with parametric uncertainty [24]. The creation of a robust signomial programming framework to capture uncertainty in engineering design, and specifically aircraft design, will allow us to have more confidence in the results of the conceptual design phase, reduce program risk, and increase overall system performance.

D. Contributions

This paper proposes a tractable robust signomial program (RSP) that we solve as a sequential robust geometric program (RGP), allowing us to implement robustness in non-log-convex problems such as aircraft design. We extend the RGP framework developed by Saab et al. [24] to SPs, and we implement it as part of an existing open-source optimization framework in Python [25,26]. We implement the RSP formulation on a conceptual aircraft design problem with over 100 variables, as defined in Ref. [27]. The benefits of RO are demonstrated both in ensuring design feasibility and performance using Monte Carlo (MC) simulations of the uncertain parameters. We further explore the benefits of RO in multiobjective optimization, and we propose a goal programming RSP formulation for risk minimization problems.

II. Mathematical Background

A. Robust Optimization

Given a general optimization problem under parametric uncertainty, we define the set of possible realizations of uncertain vector of parameters u in the uncertainty set $\mathcal{U}$. This allows us to define the problem under uncertainty below, with objective f_0 and constraints f_i over design variables x and uncertain parameters u:

$$\begin{aligned} \min \quad & f_0(x,u) \\ \text{subject to} \quad & f_i(x,u) \le 0, \quad \forall\ u \in \mathcal{U}, i = 1,\dots,n \end{aligned}$$

In the trivial case when $\mathcal{U}$ has a single element, we recover the deterministic problem where parameters u are fixed and certain. The problem of interest, however, has parametric uncertainty over continuous variables, for which $\mathcal{U}$ is a nonempty set with countably infinite members. This general problem is infinite-dimensional since it is possible to formulate an infinite number of constraints with the countably infinite number of possible realizations of $u \in \mathcal{U}$. To circumvent this issue, we can define the following robust formulation of the uncertain problem as follows:

$$\begin{aligned} \min \quad & f_0(x,u) \\ \text{subject to} \quad & \max_{u \in \mathcal{U}} f_i(x,u) \le 0, i = 1,\dots,n \end{aligned}$$

This formulation hedges against the worst-case realization of the uncertainty in the defined uncertainty set. The set is often described by a norm, which contains possible uncertain outcomes from distributions with bounded support

$$\begin{aligned} \min \quad & f_0(x,u) \\ \text{subject to} \quad & \max_{u} f_i(x,u) \le 0, i = 1,\dots,n \\ & \|u\| \le \Gamma \end{aligned} \tag{1}$$

where Γ is defined by the user as a global uncertainty bound. The larger the Γ, the greater the size of the uncertainty set that is protected against.

B. Geometric Programming

A *geometric program in posynomial form* is a log-convex optimization problem of the form:

$$\begin{aligned} \min \quad & f_0(\boldsymbol{u}) \\ \text{subject to} \quad & f_i(\boldsymbol{u}) \le 1, i = 1, \ldots, m_p \\ & h_i(\boldsymbol{u}) = 1, i = 1, \ldots, m_e \end{aligned} \tag{2}$$

where each f_i is a *posynomial*, each h_i is a *monomial*, m_p is the number of posynomials, and m_e is the number of monomials. A monomial $h(\boldsymbol{u})$ is a function of the form

$$h_i(\boldsymbol{u}) = e^{b_i} \prod_{j=1}^{n} u_j^{a_{ij}} \tag{3}$$

where a_{ij} is the jth component of a row vector $\boldsymbol{a_i}$ in $\mathbb{R}^n$, u_j is the jth component of a column vector $\boldsymbol{u}$ in $\mathbb{R}^n_+$, and b_i is in $\mathbb{R}$. An example of a monomial is the lift equation $L = (1/2)\rho V^2 C_L S$. A posynomial $f(\boldsymbol{u})$ is the sum of $K \in \mathbb{Z}^+$ monomials:

$$f_i(\boldsymbol{u}) = \sum_{k=1}^{K} e^{b_{\rm ik}} \prod_{j=1}^{n} u_j^{a_{\rm ikj}} \tag{4}$$

where $a_{\rm ikj}$ is the jth component of a row vector $\boldsymbol{a}_{ik}$ in $\mathbb{R}^n$, u_j is the jth component of a column vector $\boldsymbol{u}$ in $\mathbb{R}^n_+$, and b_{ik} is in $\mathbb{R}$ [19]. The stagnation pressure definition is a good example: $P_t = P + (1/2)\rho V^2$.

A logarithmic change of the variables $x_j = \log(u_j)$ would turn a monomial into the *exponential of an affine function* and a posynomial into the *sum of exponentials of affine functions*. A transformed monomial $h_i(\boldsymbol{x})$ is of the form

$$h_i(\boldsymbol{x}) = e^{\boldsymbol{a}_i \boldsymbol{x} + b_i} \tag{5}$$

where $\boldsymbol{x}$ is a column vector in $\mathbb{R}^n$. A transformed posynomial $f_i(\boldsymbol{x})$ is the sum of $K_i \in \mathbb{Z}^+$ monomials:

$$f_i(\boldsymbol{x}) = \sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}\boldsymbol{x} + b_{ik}} \tag{6}$$

where $\boldsymbol{x}$ is a column vector in $\mathbb{R}^n$. A geometric program with transformed constraints is a *geometric program in exponential form*, and it is a convex optimization problem.

The positivity of exponential functions restricts the space spanned by posynomials and limits GPs to certain classes of problems. However, since many engineering problems of interest have purely positive quantities, GPs are quite applicable; and certain variable transformations can make problems with negative quantities tractable. The restriction of posynomials to the less-than side of inequalities is a more significant barrier, and it motivates the introduction of signomials.

C. Signomial Programming

A *signomial* can be defined as the difference of two posynomials. Consequently, a SP is a non-log-convex optimization problem of the form

$$\begin{aligned} \text{minimize} \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & f_i(\boldsymbol{x}) - g_i(\boldsymbol{x}) \le 0, i = 1, \ldots, m \end{aligned} \tag{7}$$

where f_i and g_i are both posynomials, and $\boldsymbol{x}$ is a column vector in $\mathbb{R}^n$.

Reliably solving a SP to a local optimum has been described in Refs. [19,28]. A common solution heuristic involves solving a SP as a sequence of GPs, where each GP is a local approximation of the SP. Although signomial programming is a powerful tool, applications involving SPs are usually prone to uncertainties that have a significant effect on the solution.

III. Robust Signomial Programming

As a preview of the following sections, robust signomial programming assumes that parameter uncertainties belong to an uncertainty set and solves a reformulated design problem to find the best solution through the process shown in Fig. 2. As long as the original optimization problem is SP compatible, a tractable robust formulation of the problem exists, making this method general. We derive the intractable formulation of a RSP below.

A *SP in exponential form* is as follows:

$$\begin{aligned} \min \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & \sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}\boldsymbol{x} + b_{ik}} - \sum_{k=1}^{G_i} e^{\boldsymbol{c}_{ik}\boldsymbol{x} + d_{ik}} \le 0, \quad \forall\ i \in 1, \ldots, m \end{aligned} \tag{8}$$

where the constraints are represented as the difference of posynomials in exponential form. Let $\boldsymbol{a}_{ik}$ and $\boldsymbol{c}_{ik}$ be the $((i-1) \times m + k)$th rows of the exponents matrices $\boldsymbol{A}$ and $\boldsymbol{C}$, respectively, and let b_{ik} and d_{ik} be the $((i-1) \times m + k)$th elements of the coefficients vectors $\boldsymbol{b}$ and $\boldsymbol{d}$, respectively.

The data $(\boldsymbol{A}, \boldsymbol{C}, \boldsymbol{b}, \boldsymbol{d})$ are assumed to be uncertain and living in an uncertainty set $\mathcal{U}$, where $\mathcal{U}$ is parametrized affinely by a perturbation vector ζ:

$$\mathcal{U} = \left\{ [\boldsymbol{A}; \boldsymbol{C}; \boldsymbol{b}; \boldsymbol{d}] = [\boldsymbol{A}^0; \boldsymbol{C}^0; \boldsymbol{b}^0 \boldsymbol{d}^0] + \sum_{l=1}^{L} \zeta_l [\boldsymbol{A}^l; \boldsymbol{C}^l; \boldsymbol{b}^l; \boldsymbol{d}^l] \right\} \tag{9}$$

where $\boldsymbol{A}^0, \boldsymbol{C}^0, \boldsymbol{b}^0$, and $\boldsymbol{d}^0$ are the nominal exponents and coefficients; $\{\boldsymbol{A}^l\}_{l=1}^L$, $\{\boldsymbol{C}^l\}_{l=1}^L$, $\{\boldsymbol{b}^l\}_{l=1}^L$, and $\{\boldsymbol{d}^l\}_{l=1}^L$ are the basic shifts of the exponents and coefficients; and ζ_l is the lth component of ζ belonging to a perturbation set $\mathcal{Z} \in \mathbb{R}^L$ such that

$$\mathcal{Z} = \{\zeta \in \mathbb{R}^L : \|\zeta\| \le \Gamma\} \tag{10}$$

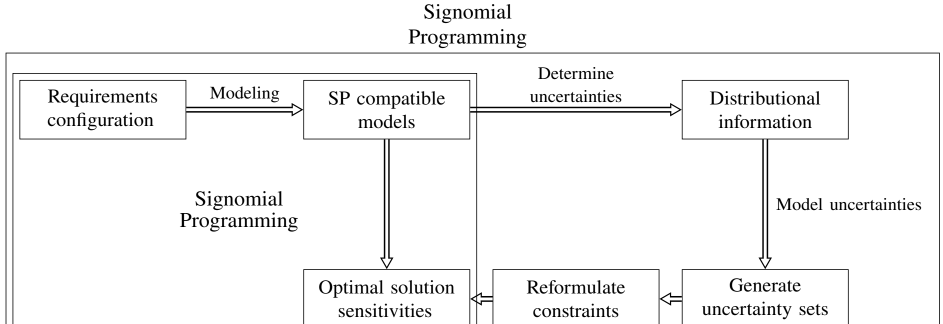


Fig. 2 Block diagram showing the difference between the design process using a SP and a RSP.

As mentioned previously, our goal is a formulation that is immune to uncertainty in the data. Accordingly, the robust counterpart of the uncertain SP in Eq. (8) is

$$\begin{aligned} &\min \quad f_0(\boldsymbol{x}) \\ &\text{subject to} \quad \max_{\zeta\in\mathcal{Z}}\left\{\sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}(\zeta)\boldsymbol{x}+b_{ik}(\zeta)} - \sum_{k=1}^{G_i} e^{\boldsymbol{c}_{ik}(\zeta)\boldsymbol{x}+d_{ik}(\zeta)}\right\} \le 0 \\ &\qquad\qquad \forall\ i\in 1,\dots,m \end{aligned} \tag{11}$$

The optimization problem in Eq. (11) is intractable using current solvers; therefore, a heuristic approach to solving RSPs approximately as a sequential RGP will be presented in the following sections. As our approach is based on robust geometric programming, a brief review of the subject will follow based on Ref. [24].

IV. Robust Geometric Programming

This section presents a brief review of the approximation of an RGP as a tractable optimization problem as discussed in Ref. [24]. The robust counterpart of an uncertain geometric program is

$$\min \quad f_0(\boldsymbol{x}) \quad \text{subject to} \quad \max_{\zeta\in\mathcal{Z}}\left\{\sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}(\zeta)\boldsymbol{x}+b_{ik}(\zeta)}\right\} \le 1, \quad \forall\ i\in 1,\dots,m \tag{12}$$

which does not have an tractable exact reformulation due to its co-NP hardness [29].

A. Simple Conservative Formulation

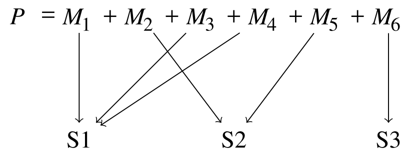
One way to approach the intractability in Eq. (12) is to replace each constraint by a tractable approximation. Replacing the maximum of the sum in Eq. (12) by the sum of the maximum will lead to the following formulation:

$$\begin{aligned} &\min \quad f_0(\boldsymbol{x}) \quad \text{subject to} \\ &\sum_{k=1}^{K_i} \max_{\zeta\in\mathcal{Z}}\{e^{\boldsymbol{a}_{ik}(\zeta)\boldsymbol{x}+b_{ik}(\zeta)}\} \le 1, \quad \forall\ i\in 1,\dots,m \end{aligned} \tag{13}$$

Maximizing a monomial term is equivalent to maximizing an affine function; therefore. Eq. (13) is tractable.

B. Equivalent Intermediate Formulation

This formulation is equivalent to the formulation in Eq. (12) but with smaller, easier to handle posynomial constraints. By the properties of inequalities, the posynomial P in posynomial inequality $M \ge P$ can be divided into an equivalent set of smaller posynomials based on the dependence between its monomial terms. Figure 3 shows how a constraint can be represented as an equivalent set of smaller posynomial constraints.



$$\max\{P\} \le 1 \iff \begin{aligned} t_1 + t_2 + t_3 &\le 1 \\ \max\{S_1\} = \max\{M_1 + M_3 + M_4\} &\le t_1 \\ \max\{S_2\} = \max\{M_2 + M_5\} &\le t_2 \\ \max\{S_3\} = \max\{M_6\} &\le t_3 \end{aligned}$$

Fig. 3 Partitioning of a large posynomial into smaller posynomials, requiring the addition of auxiliary variables. S_i are posynomials with independent sets of variables.

The posynomial constraints are categorized into three sets: large posynomials, two-term posynomials, and monomials, represented by S_1, S_2, and S_3, respectively. Monomials are tractable, and two-term posynomials can be well approximated using piecewise-linear functions [30]. We implement the following two tractable approximations for large posynomials.

1. Linearized Perturbations Formulation

If the exponents are known and certain, then uncertain large posynomial constraints can be approximated as signomial constraints. The exponential perturbations in each posynomial are linearized using a modified least squares method, and then the posynomial is robustified using techniques from robust linear programming. The resulting set of constraints is SP compatible, therefore, a RGP can be approximated as a SP.

2. Best Pairs Formulation

If the exponents are also uncertain, then large posynomials cannot be approximated as a SP, and further simplification is needed. This formulation aims to maximize each pair of monomials in each posynomial while finding the best combination of monomials that gives the least conservative solution. Reference [24] provides the descent algorithm used to find such locally optimal monomial combinations. Additionally, it demonstrates that a GP with uncertain exponents can be approximated as a GP for polyhedral uncertainty, and a conic optimization problem for ellipsoidal uncertainty. For a specific description of the descent method and the form of approximate RGP formulations, please refer to [24].

V. Approach to Solving Robust Signomial Programs

This section presents a heuristic algorithm to solve a RSP based on our previous discussion on robust geometric programming.

A. General RSP Solver

As mentioned in Sec. II.C, a common heuristic algorithm to solve a SP is by sequentially solving local GP approximations. Similarly, our approach to solve a RSP is based on solving a sequence of local RGP approximations. In Fig. 4, we provide a step-by-step algorithm. In this heuristic, a good initial guess will lead to faster convergence and possibly a better solution. The solution of the SP without uncertainty is in general a good candidate x_0.

For comparisons between methods ahead, we write the algorithm explicitly as follows:

1) Choose an initial guess x_0.
2) Repeat:
   a) Find the local GP approximation of the SP at x_i.
   b) Find the RGP formulation of the GP.
   c) Solve the RGP to obtain x_{i+1}.
   d) If $x_{i+1} \approx x_i$, break.

Any of the previously mentioned methodologies can be used to formulate the local RGP approximation. However, depending on the RGP formulation chosen to solve an RSP, the formulation and solution blocks in Fig. 4 are adjusted.

B. Best Pairs RSP Solver

If the best pairs methodology is exploited, then the preceding algorithm would change so that each iteration would solve the local RGP approximation and choose the best permutation for each large posynomial. The modified algorithm is then as follows:

1) Choose an initial guess x_0.
2) Repeat:
   a) Find the local GP approximation of the SP at x_i.
   b) For each large posynomial constraint, select the new permutation ϕ such that ϕ minimizes the robust large constraint evaluated at x_i.
   c) Solve the approximate tractable counterparts of the local GP in Eq. (12), and let $\boldsymbol{x}_{i+1}$ be the solution.
   d) If $x_{i+1} \approx x_i$, break.

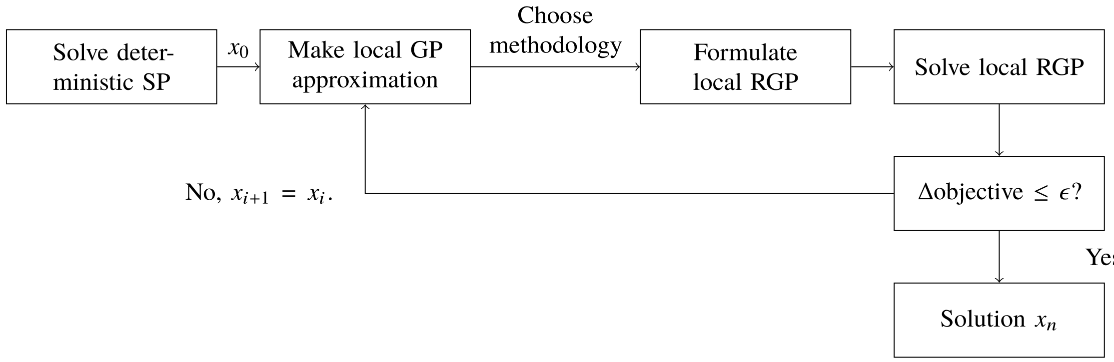


Fig. 4 Block diagram showing the steps of solving a RSP.

C. Linearized Perturbations RSP Solver

If the linearized perturbations formulation is to be used, there is an additional challenge in that the robust counterpart of each local GP is a SP. To approximately solve the resulting SP at each iteration, the signomial constraints in each robust approximation are themselves locally approximated by GP-compatible constraints. The algorithm then becomes as follows:

1) Choose an initial guess x_0.
2) Repeat:
   a) Find the local GP approximation of the SP at x_i.
   b) Robustify the constraints of the local GP approximation using the linearized perturbations methodology.
   c) Find the local GP approximation of the resulting local SP at x_i.
   d) Solve the local GP approximation in step c to obtain x_{i+1}.
   e) If $x_{i+1} \approx x_i$, break.

VI. Models

We implement the aforementioned RSP formulation on an unmanned gas-powered aircraft design problem that is systematically developed in Ref. [27] with the ellipsoidal fuselage model borrowed from Ref. [21]. We optimize a wing, fuselage, and engine given a payload and range requirement. The optimization model was developed using GPkit (a Python package that provides abstractions for using GPs in engineering design [26]) and captures fundamental tradeoffs in aircraft design. The nominal model has 175 variables and 153 constraints: a common level of sparsity for GP and SP models. A short qualitative overview of the model follows; for more detailed information, please refer to Refs. [21,27]. The uncertainties associated with the parameters will be described in Sec. VII.

A. Flight Profile

The flight profile model is borrowed from Ref. [3]. Within the model, the trajectory of the aircraft is optimized over four steady flight segments.

B. Atmosphere

The atmosphere model is taken from Ref. [31] and considers changes in density and dynamic viscosity with altitude for a standard atmosphere.

C. Aircraft

The aircraft is modeled as a wing, fuselage, and engine system. The aircraft is assumed to be in steady flight so that the thrust power is equal to the sum of the drag power and rate of change of potential energy of the aircraft; and the lift is equal to the total weight, ignoring the vertical component of thrust in climb. Its total weight is the sum of its components. The aircraft has to be able to take off at specified minimum speed without stalling as well. Aircraft component models are detailed as follows.

1. Wing

Lift is generated by the wing as a function of its geometry and freestream conditions. The wing structure model is based on a beam model with a distributed lift load and a point mass in the center representing the fuselage. Wing fuel volume is modeled as a fraction of the internal volume available in the wing. The weight of the wing is the sum of skin and spar weights. Its drag is the sum of induced and profile drags, the latter of which is constrained by a three-term softmax-affine posynomial fit [32] of drag polars generated in XFOIL [33]. The airfoil used was designed by Professor Mark Drela of Massachusetts Institute of Technology and is a variant of those implemented in Ref. [21].

2. Fuselage

The fuselage contains the fuel and payload internally, and it contains the engine externally. It is assumed to be ellipsoidal in shape, and its drag is estimated using a form factor. The fuselage is assumed not to contain any structural members, and so its weight consists only of skin weight.

3. Engine

The aircraft is powered by a naturally aspirated piston engine. It is subject to power lapse at lower air densities at higher altitudes. Engine weight versus maximum sea level power as well as brake specific fuel consumption versus thrust and altitude are modeled using the posynomial fits of engine performance data from Ref. [34].

D. Source of Non-Log Convexity: Fuel Volume

The fuel models have been detailed in the previous sections, but it is noteworthy that the signomial constraint in the optimization appears in the aircraft total fuel volume constraint, as shown in Eq. (14):

$$V_f \leq V_{f_{\text{wing}}} + V_{f_{\text{fuse}}} \tag{14}$$

The signomial constraints makes the problem non-log convex, which means that the solution methods detailed by Saab [24] need to be extended to accommodate this optimization problem.

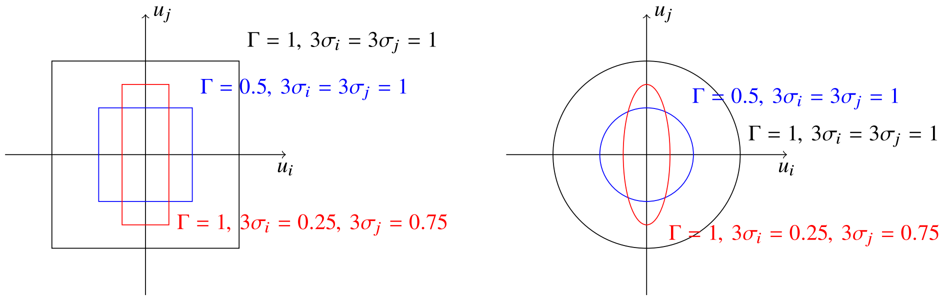
VII. Uncertainties and Sets

As mentioned in Sec. I.B, one of the advantages of RO over SO is the fact that it only requires as inputs the uncertainty set size instead of complete probability distributions over each parameter. In the context of this work, the set size is defined relatively in each uncertain design parameter u_i by $3\sigma_i$, and it is defined globally by scalar parameter Γ. An illustration of the relationship between 3σ's and Γ is provided in Fig. 5, and it is explained in Secs. VII.A and VII.B.

A. Design Parameter Uncertainties

The relative size of the uncertainty set in each uncertain variable is given by three times the coefficient of variation (CV),** as listed in

**The CV is defined as follows: CV $= \sigma/|\mu|$, where is the standard deviation and is the mean of the parameter.



a) Example L∞ or box uncertainty sets **b) Example L2 or elliptical uncertainty sets**

Fig. 5 Γ defines the overall size of norm uncertainty sets, whereas 3σ defines the relative size of the set in each uncertain parameter.

Table 1. Since for the rest of this work all standard deviations σ are normalized by the means of the parameters, we will use 3σ to represent three CVs.

In this case of a conceptual aircraft design with no prior data, the parameter uncertainties reflect aerospace engineering intuition. The wing weight coefficients $W_{\mathrm{coeff,strc}}$ and $W_{\mathrm{coeff,surf}}$, as well as the ultimate load factor N_{ult}, have large 3σ because the build quality of aircraft components is often difficult to quantify with a large degree of certainty. The payload weight and density (W_p and ρ_p) have large uncertainties since the payload is often developed concurrently with the aircraft. Parameters that engineers take to be physical constants (sea level air viscosity and density, μ and ρ) and those that can be determined with a relatively high degree of accuracy (e) have relatively low deviations. Parameters that require testing to determine ($C_{L,\max}$, $C_{f,\mathrm{ref}}$, and $V_{\min}$) have a level of uncertainty that reflects the expected variance of empirical studies. However, note that these quantities are ultimately picked by the designer using prior experience and data, and the level of conservativeness in the design will be greatly affected by the chosen 3σ.

B. Uncertainty Sets Considered

The robust design problem is solved for box and ellipsoidal uncertainty sets, which are defined by the L∞ and L2 norms, and bounded by varying the parameter Γ. Intuitively, for both sets, Γ is a global measure of how much risk is being hedged against; it affects all parameter uncertainties simultaneously. $\Gamma = 0$ implies that all of the parameters take their nominal values with zero uncertainty, which we call the nominal problem, and larger Γ protects against greater uncertainty. Γ is more rigorously defined in the context of robust linear programming in the Appendix.

For box uncertainty, Γ scales the width of the L∞ hypercube as shown in Fig. 5a, whose dimensionality is the same as the number of uncertain parameters (which is 12). More intuitively, $\Gamma \times 3\sigma_i$ defines the range of the possible values of uncertain parameter u_i normalized by the mean of u_i. It can be easy to assume that using margins and box uncertainty sets will yield the same designs, but they fundamentally function differently. First, the worst-case outcome in box uncertainty can come from the interior of the uncertainty set instead of the corner of the hypercube considered by margins. Furthermore, there is no guarantee (and it is unlikely) that the chosen corner (i.e., particular allocation of margins) is the most conservative point in the uncertainty set. It is even possible that the wrong sign of margin is allocated for certain parameters since SPs are nonlinear and local sensitivities cannot be used reliably to intuit global behavior. Consider in this particular example the sea level air density ρ. Higher air density is better for takeoff performance and naturally aspirated engine performance, but it results in higher drag; so, it is difficult for a designer to determine how to best allocate margin on ρ. Thus, for the rest of this paper, the direction of margins is determined using the local sensitivities of the nominal solution, which are obtained at no extra computational cost in the solution of the terminal GP approximation of the SP. With these considerations in mind, box uncertainty is expected to be strictly more conservative and more appropriate than the use of margins in conceptual design since 1) margins fail to capture the level of conservativeness they signal, and 2) prior information (in this case, the nominal solution) is required to allocate margin effectively.

For ellipsoidal uncertainty, Γ is the maximum diameter of the Euclidian norm ball of u as shown in Fig. 5b, where u_i is one-third the number of standard deviations of perturbation of the ith parameter from its nominal value. Ellipsoidal uncertainty exploits the fact that the joint probability of multiple uncertain parameters taking values in the tails of their respective distributions is very low. So, although it does not protect deterministically for all outcomes of the uncertain parameters within 3σ, it is expected to protect against uncertain outcomes less conservatively than the box uncertainty set, with little compromise in the ability of the design to satisfy constraints.

VIII. Results

We implement our RSP algorithm on the aforementioned conceptual aircraft design problem. Our objective function is total fuel

Table 1 Parameters and uncertainties (increasing order)

Parameters	Description	Value	Uncertainty (3σ), %
e	Span efficiency	0.92	3
μ	Air viscosity (SL)	1.78×10^{-5} kg/(m · s)	4
ρ	Air density (SL)	1.23 kg/m^3	5
$C_{L,\max}$	Stall lift coefficient	1.6	5
k	Fuselage form factor	1.17	10
$C_{f,\mathrm{ref}}$	Reference fuselage skin-friction factor	0.455	10
ρ_p	Payload density	1.5 kg/m^3	10
N_{ult}	Ultimate load factor	3.3	15
$V_{\min}$	Takeoff speed	35 m/s	20
W_p	Payload weight	3000 N	20
$W_{\mathrm{coeff,strc}}$	Wing structural weight coefficient	2×10^{-5} 1/m	20
$W_{\mathrm{coeff,surf}}$	Wing surface weight coefficient	60 N/m^2	20

SL = sea level.

consumption, which is to be minimized given a payload and range requirement.

A. Mitigation of Probability of Failure

First, the optimization problem is solved in presence of no uncertainty. Then, using the sign of sensitivities of the nominal solution, we assign 3σ margins for each parameter and generate a design using margins. These two solutions are compared with RO results for box and ellipsoidal uncertainty sets at $\Gamma = 1$ using the best pairs robustification method. From here onward, we refer to aircraft designed under no uncertainty, under margins, under box uncertainty, and under ellipsoidal uncertainty as "the nominal aircraft," "the margin aircraft," "the box aircraft," and "the ellipsoidal aircraft," respectively.

The design variables are then fixed for each solution, and the designs are simulated for different realizations of the uncertain parameters. This allows for statistical analysis of design performance and an estimate of each design's probability of constraint violation, which we define as its probability of failure (POF). In this MC scheme, the random variables are simulated from independent and identically distributed 3σ truncated Gaussians. We simulate from the truncated Gaussian since this makes it possible to confirm mathematically that for $\Gamma = 1$, all simulations of 3σ uncertain parameters are deterministically feasible for the box uncertainty set. The results are in Table 1. Designs for each solution for the rest of the section are simulated with the same MC samples for consistency.

It is noteworthy in the POF at the bottom of Table 2 that, for the nominal problem ($\Gamma = 0$), only 12% of the MC evaluations result in feasible solutions. This means that an aircraft designed for the average case would almost surely fail to satisfy the mission requirements, even with equal likelihood of favorable versus unfavorable uncertain outcomes from the symmetric truncated Gaussian. That being said, depending on the problem, it may be necessary to sacrifice performance to achieve a high degree (3σ) of reliability as in the solution for $\Gamma = 1$. Furthermore, the margin aircraft, the box aircraft, and the ellipsoidal aircraft spend on average 53, 55, and 39% more fuel, respectively, than the aircraft designed for the nominal case; but they also are robust to all uncertain outcomes in the 3σ set for the MC simulation.

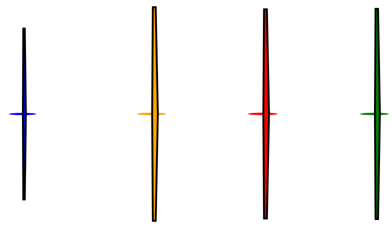
Table 2 also indicates that margins are not a good method of allocating uncertainty. The claim for the use of margins is that they protect against the worst-case outcome of each parameter, but the results show otherwise. Since the box design at $\Gamma = 1$ is strictly more conservative (worse worst-case outcome) over the 3σ hypercube than the margin design, we see that a margin from the interior of the hypercube rather than its corner is more effective in protecting against the worst case. Furthermore, there are no probabilistic guarantees that the aircraft with margins would not fail one of the MC simulations. Given enough samples, it is almost surely true that some MC simulations will violate feasibility for the design with margins, whereas box uncertainty guarantees deterministically that the constraints are satisfied.

We also posited that the ellipsoidal uncertainty, although it does not protect deterministically against all 3σ uncertainties, would be less conservative than the margin and box designs while not significantly sacrificing POF. This is confirmed since the ellipsoidal design fails none of the random samples, and it spends 9% and 10% less fuel on average than the margin and box aircraft, respectively. The significance of this cannot be understated: the use of ellipsoidal uncertainty results in designs that have strictly better performance outcomes while protecting against a similar amount of risk as designs using margins or box uncertainty.

An analysis on the range $\Gamma = [0, 1]$ was performed to confirm that the trends from Table 2 hold for all Γ. Figure 6 shows that POF goes monotonically toward zero as Γ increases for all three methods, where box uncertainty is more conservative than ellipsoidal uncertainty over the whole Γ domain, with no such guarantees for margins.

In absolute terms, the nominal SP under zero uncertainty or with margins takes just under 0.9 s to solve on a modern laptop computer

Table 2 SP aircraft optimization results for $\Gamma = 1$

	Description	Units	No uncertainty	Margins	Box	Ellipsoidal
Free variables						
L/D	Mean lift-to-drag ratio	——	45.0	35.4	36.1	38.4
AR	Aspect ratio	——	38.0	25.0	24.6	28.1
Re	Reynolds number	——	8.44×10^5	1.21×10^6	1.35×10^6	1.21×10^6
S	Wing planform area	m^2	6.27	14.9	14.6	12.8
τ	Airfoil thickness ratio	——	0.175	0.197	0.198	0.192
V	Mean flight velocity	m/s	41.7	34.6	35.4	36.2
T_{flight}	Time of flight	h	20.0	24.1	23.6	23.1
W_w	Wing weight	N	1170	2080	2090	1940
$W_{w,\text{strc}}$	Wing structural weight	N	792	1100	1130	1090
$W_{w,\text{surf}}$	Wing skin weight	N	376	985	966	851
W_{fuse}	Fuselage weight	N	151	192	177	168
W_e	Engine weight	N	84.4	111	122	115
$V_{f,\text{avail}}$	Total fuel volume	m^3	0.0267	0.0458	0.0502	0.0459
$V_{f,\text{fuse}}$	Fuselage fuel volume	m^3	0.0134	0	0	0
$V_{f,\text{wing}}$	Wing fuel volume	m^3	0.0133	0.0680	0.0667	0.0468
	Sketches to scale					
Metrics						
Objective	Total fuel weight	N	214	367	402	368
E[Objective]	Expected total fuel weight	N	207	316	320	287
σ[Objective]	std. dev. of fuel weight	N	11	12	12	11
P[failure]	Probability of failure	%	88	0	0	0

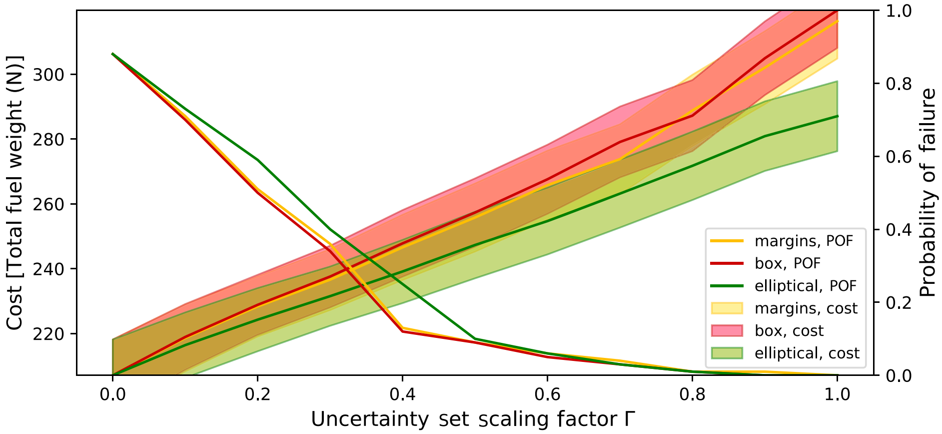


Fig. 6 Simulated cost and POF of optimal margin, box, and ellipsoidal aircraft as a function of Γ. Banded lines represent mean and standard deviation of total fuel burn, simulated with 100 MC samples of uncertain parameters.

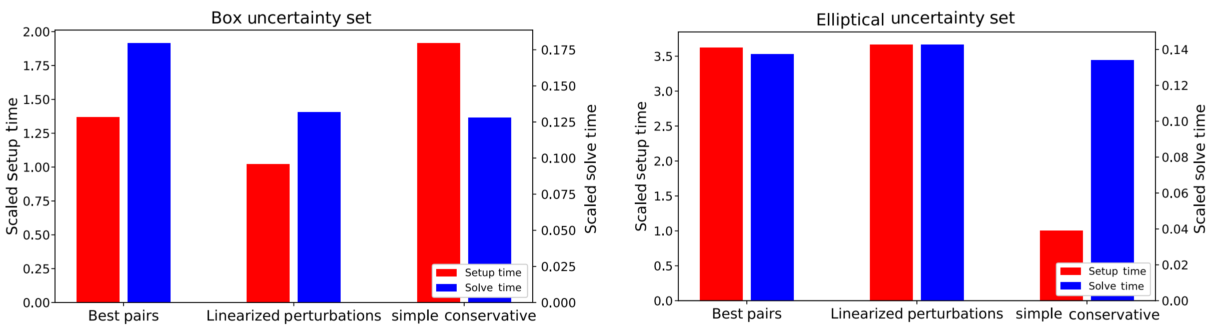


Fig. 7 Robust aircraft optimization setup and solution times, normalized by the nominal problem solution time, for Γ = 1.

using MOSEK [35], which is an interior point solver that is free for academic use; the authors refer to Refs. [3,36] for more in-depth SP solution time analyses. In Fig. 7, we examine briefly in relative terms how the different RSP methodologies compare in terms of setup and run times. Since the setup time of the nominal problem is minimal, we have normalized the results by the solution time of the nominal problem. The bottom axis ranks the methods by their level of conservativeness, with best pairs and simple conservative formulations being the least and most conservative, respectively, and where the ellipsoidal formulations are less conservative than the box formulations. For this aircraft design problem, the preferred best pairs methodology with an ellipsoidal uncertainty set is competitive in solution and setup times relative to other methods while providing the least conservative solutions. Note that setup and solution times for RSPs are highly problem specific, and so it is not possible to predict the time performance of other RSP-compatible problems from these results. Time performance will vary, depending on the number of inequality constraints, the degree of coupling between monomials in each inequality, and the RGP approximation and uncertainty set used.

B. Effect of Robustness on Multiobjective Performance

One of the benefits of convex and difference-of-convex optimization methods is the ability to optimize for different objectives [3]. As a demonstration, we optimize the aircraft without uncertainty for six different objectives, and we show the nondimensionalized figures of merit in Table 3. Since the model is physics based, the model can even accommodate objectives such as wing area, which are often unintuitive and not considered. The resulting aircraft differ significantly with respect to certain objectives while being similar in many others. As an example, takeoff weights for all aircraft are 0.87 to 1.22 times the baseline total fuel solution, whereas engine weights vary from 0.88 to 3.62 times the baseline. These demonstrate the importance of considering many objectives in design, and they underline the power of SPs in helping consider the multiobjective performance of engineered systems.

We demonstrate the benefits of RSPs in multiobjective optimization by considering uncertainty while optimizing for the same objectives. We perform the optimization of the aircraft with no uncertainty, and both box and ellipsoidal uncertainty ($\Gamma = 1$) for the objective functions in Table 3. The resulting aircraft are sketched in Fig. 9, and their objective performance is plotted on radar plots in Fig. 8. Radar plots are useful because they allow engineers to visualize the performance of designs in many dimensions. One way to envision the multiobjective performance of the aircraft is to consider the area of the polygon defined by the aircraft's performance as the figure of merit; the smaller, the better.

Figure 8 shows the effects of robustness on the different worst-case performance metrics of the different aircraft. As expected, the box

Table 3 Nondimensionalized variations in objective values with respect to the aircraft optimized for different objectives[a]

Objective	Total fuel	Total cost	Takeoff weight	1/(cruise L/D)	Engine weight	Wing area
Total fuel	1.00	1.00	1.00	1.00	1.00	1.00
Total cost	2.51	0.64	0.98	2.55	3.62	0.98
Takeoff weight	1.37	0.89	0.87	1.43	1.42	0.87
1/(cruise L/D)	1.29	1.00	1.11	0.77	2.48	1.53
Engine weight	1.35	1.53	1.22	1.40	0.88	2.78
Wing area	1.37	0.89	0.87	1.43	1.43	0.87

[a]Objective values are normalized by the total fuel solution.

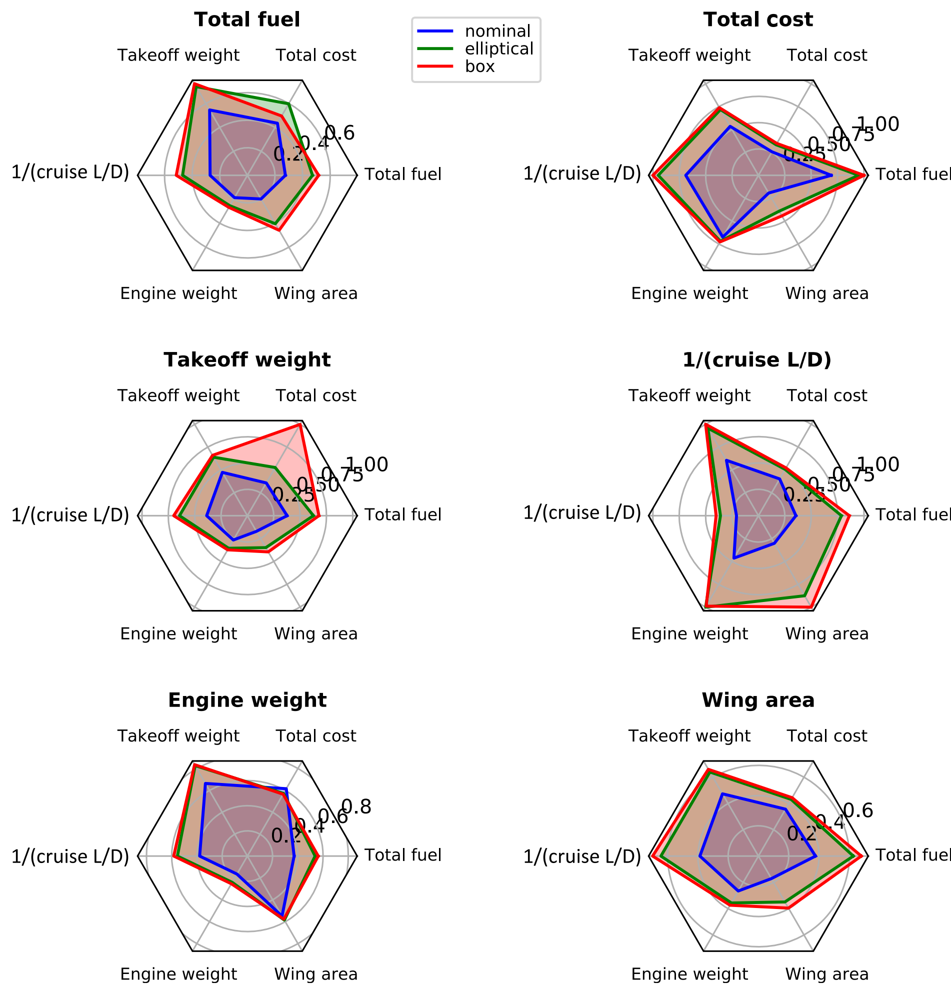


Fig. 8 Radar plots of aircraft performance. Bolded titles are optimized objectives for each plot, and individual plots show nondimensionalized multiobjective performance of aircraft, designed under different uncertainty sets.

uncertainty set is strictly more conservative than the ellipsoidal uncertainty set in the optimized objective. However, the tradeoffs in the others are less clear. Note that the radar plots show the worst-case performance of the vehicles, although this analysis can also be performed for the mean performance of the aircraft determined through MC simulation.

This multiobjective comparison underscores the sensitivity of different objectives to level of robustness and by extension parameter uncertainty. For example, the engine weight of the 1/(cruise L/D) solution is highly sensitive to level of robustness, whereas the engine weight of the total (time and fuel) cost of the aircraft is insensitive. Therefore, we might want to consider total cost to be our overall objective instead of 1/(cruise L/D) if we are relatively averse to risk in engine versus airframe design. Robustness can affect the efficacy of different choices of objective function in ensuring multiobjective performance. Since RSPs can be solved quickly and reliably over a variety of objective functions, they allow engineers to understand these kinds of complex tradeoffs early on in the design process.

Based on these observations, we argue that there could be significant value left on the table if uncertainty is not considered with sufficient mathematical rigor in early phases of the design process. RSPs allow engineers to capture complex tradeoffs in nonlinear optimization problems while considering uncertainty, resulting in less conservative solutions than solutions that implement margins and other less mathematically rigorous methods for risk mitigation. Thus, RSPs improve significantly on the paradigms of design under uncertainty in use in the aerospace industry today.

C. Risk Minimization Problems

All of the previous multiobjective analyses have assumed that we have an understanding of exactly the amount of uncertainty we are willing to tolerate. However, minimizing risk can also be the objective of our model. This would suggest the following formulation:

$$\begin{aligned} \max \quad & \Gamma \\ \text{subject to} \quad & f_i(x,u) \le 0, \quad i = 1,\dots,n \\ & \|u\| \le \Gamma \\ & f_0(x) \le (1+\delta) f_0^*, \quad \delta \ge 0 \end{aligned} \tag{15}$$

where f_0^* is the optimum of the nominal problem in formulation (1); and δ is a fractional penalty on the objective that we are willing to sacrifice for robustness, which gives $(1+\delta)f_0^*$ as the upper bound on the objective value. Intuitively, this is a form of goal programming, where we specify the exact maximum worst-case value of an objective we can tolerate with the goal of maximizing the size of the uncertainty set we can handle.

The goal programming problem in formulation (15) is clearly not equivalent to the problem in formulation (1) but should yield the same results if there is no optimality gap between the methods. To show this, we use the worst-case objective values from the POF study shown in Fig. 6 as the δ inputs to the goal programming model, and we compare the results. The results are presented in Table 4. Note that the two methods were evaluated MC runs using the same 100 realizations of the uncertainty for consistency in POF results.

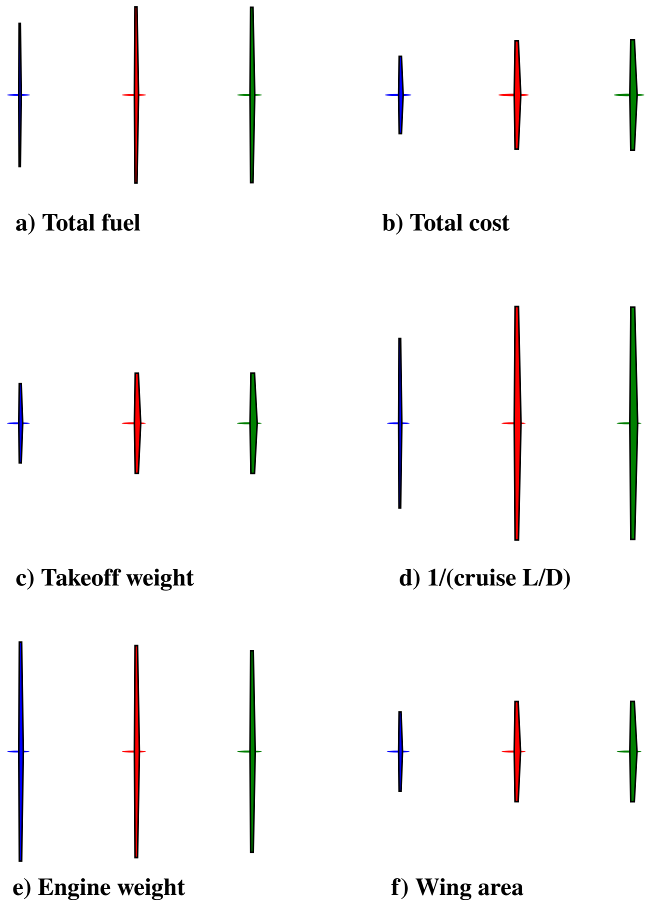


Fig. 9 Sketches of the aircraft drawn for corresponding radar plots. Drawn to scale for comparison.

First, note that there are no results reported for the goal program for zero uncertainty: $\Gamma = [0.00]$. Since the feasible set of this problem is a point design, the signomial program solution heuristic declares the problem infeasible after being unable to locate the singular feasible region. However, when we positively perturb the singular δ, the goal program has a nonempty feasible set and returns the same solution as the original RO method. Otherwise, the Γ values found by the goal program match exactly with the original RO problem. We confirm that both methods produce the same designs by examining the physical dimensions of the aircraft, as well as through the probability of failure found through MC simulation in Table 4. Note that there are small discrepancies in the POF: notably, in the values for

Table 4 Results of original RO problem versus goal program in terms of size of uncertainty set Γ, objective penalty δ, and POF[a]

RO form			Goal form		
Γ	δ	POF	δ	Γ	POF
0.00	7.95×10^{-5}	0.88	7.95×10^{-5}	——	——
0.10	0.0525	0.73	0.0525	0.10	0.73
0.20	0.108	0.59	0.108	0.20	0.59
0.30	0.168	0.40	0.168	0.30	0.39
0.40	0.231	0.25	0.231	0.40	0.25
0.50	0.298	0.10	0.298	0.50	0.10
0.60	0.370	0.06	0.370	0.60	0.07
0.70	0.447	0.03	0.447	0.70	0.03
0.80	0.519	0.01	0.519	0.80	0.01
0.90	0.618	0.00	0.618	0.90	0.00
1.00	0.714	0.00	0.714	1.00	0.00

[a]Both methods use the best pairs formulation under ellipsoidal uncertainty. The designs obtained through the two different methods match.

$\Gamma = [0.3, 0.6]$. This is possible because there are uncertainty realizations that can fall in or out of feasibility due to numerical precision. The interior point solvers used cannot make computations exactly.

We can also expand this framework to perform multivariate goal programming by changing formulation (15) to include all objectives we are interested in:

$$f_{0,j}(x) \leq (1 + \delta_j) f^*_{0,j}, \quad \delta_j \geq 0, \quad j = 1, \ldots, m \tag{16}$$

The benefit of goal programming is that it allows us to explore multidisciplinary tradeoffs without having to enumerate the design space along each objective direction. The term multiobjective optimization is misleading because you can only optimize for one objective at once. The design is going to be influenced by how engineers weigh different objectives, and it is not obvious whether an objective should be a constraint instead. The most fundamental choice that an engineer can make in design is what the objective function is, and it is often the case that there are many potential objectives that are conflicting. But, risk is ubiquitous in engineering design problems, and so goal programming allows risk to be used as a global design variable against which all objectives can be weighed.

IX. Potential Future Work or Studies

There are a myriad of potential extensions to signomial programming under uncertainty. In the spirit of helping reduce program risk in aerospace design, the authors make a few observations and recommendations.

In this study, we do not discriminate between the kinds of constraints violated. However, it would be possible to rank the severity of constraint violations so as to penalize some (e.g., structural safety) more heavily than others (maximum range constraint). This would inject further realism into design under uncertainty since some violations contribute to program risk more significantly than others.

Another potentially valuable extension to the proposed framework is the concurrent implementation of multiple sets to contain the uncertain parameters, with the purpose of restricting uncertain outcomes further and reducing conservativeness. One example of this would be to impose an L0 norm as well as an L2 norm to bound the size of uncertainty set. This method can be used to set the total size of the uncertainty set in a Euclidian sense but then also to restrict the uncertainty to a subset of all of the uncertain parameters. This also turns the problem into a mixed-integer robust optimization problem, which poses interesting computational challenges.

With respect to interesting studies, RO opens up the possibility to discover and analyze with mathematical rigor the benefits of adaptable architectures in aircraft design versus more traditional point designs. Some examples of these are modular designs, morphing designs, adaptively manufactured designs, and aircraft families. It is likely that these types of engineered robustness become more effective at reducing program risk in presence of uncertainty since they are more likely to deliver value under adverse stochastic outcomes.

In situations where there are data available to aid design, RO can help explore the design space while taking into account the sparsity of and noise in the data. This opens up an array of potential trade studies where engineers can learn about the exposure of designs to the quality of data and attempt to gather data that best reduce the uncertainty in the performance of designs.

X. Conclusions

This paper has motivated the use of RSPs in conceptual engineering design in lieu of the mathematically nonrigorous methods of optimization under uncertainty widely used in the aerospace industry today. A tractable RSP formulation has been developed in response to a need to optimize over uncertain parameters, extending an existing tractable approximate RGP framework to non-log-convex problems. This RSP formulation is a valuable contribution to the fields of robust optimization and difference-of-convex programming.

RSPs have a wide variety of potential applications in engineering design. The use of RSPs in conceptual design is demonstrated to result in systems that are more robust to uncertainties in operational parameters, such as payload mass and range, as well as uncertain environmental and manufacturing parameters. Unlike legacy methods, this robustness has probabilistic guarantees, where sets of size $\Gamma = 1$ protect against all realizations of uncertainty for a given set of parameters. Thus, engineers can use robust signomial programming to trade off robustness and optimality within engineered systems in a tractable and mathematically rigorous manner.

Robust designs under box and ellipsoidal uncertainties were compared to a design implementing margins. It was confirmed that designs using box uncertainty are strictly more conservative than designs using margins. This indicates that the traditional method of allocating margins by observing the local sensitivities of the nominal solution is inadequate since it does not represent the worst-case outcomes of 3σ uncertain parameters as claimed. Furthermore, it is shown that box uncertainty has approximately the same expectation and standard deviation as the solution with margin but provides probabilistic guarantees of feasibility, unlike its counterpart.

It was also confirmed that ellipsoidal designs are strictly less conservative than margin or box designs while protecting against the same parametric uncertainties. Since designs found using RSP under ellipsoidal uncertainty are less conservative than designs found through traditional methods, RSPs have the potential to reduce the program risk and increase the performance of designs compared to traditional methods with no sacrifice in reliability.

RO has the potential to change current aerospace design paradigms by introducing mathematical rigor to design under uncertainty. Current aerospace conceptual design practices still rely heavily on the expertise of established engineers, even in the absence of prior experience exploring the design space. RSPs provide new opportunities in aerospace conceptual design since they are compatible with physics-based models that are deprived of or lacking in data, and they bring quantitative measures of design reliability to the table.

Appendix: Robust Linear Programming: A Quick Review

As mentioned earlier, principles from robust linear programming are used formulate an approximate robust geometric program.

Consider the system of linear constraints

$$\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \leq 0$$

where

$$\begin{aligned} &\boldsymbol{A} \text{ is } m \times n \\ &\boldsymbol{x} \text{ is } n \times 1 \\ &\boldsymbol{b} \text{ is } m \times 1 \end{aligned} \tag{A1}$$

where the uncertain data are contained in a set defined by Eqs. (9) and (10).

A.1. Box Uncertainty Set

If the perturbation set $\mathcal{Z}$ given in Eq. (10) is a box uncertainty set (i.e., $\|\zeta\|_\infty \leq \Gamma$), then the robust formulation of the ith constraint is equivalent to

$$\Gamma \sum_{l=1}^{L} |-b_i^l - \boldsymbol{a}_i^l \boldsymbol{x}| + \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 \leq 0 \tag{A2}$$

If only b is uncertain (i.e., $A^l = 0, \forall l = 1, 2, \ldots, L$), then Eq. (A2) becomes

$$\sum_{l=1}^{L} \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 + \Gamma \sum_{l=1}^{L} |b_i^l| \leq 0 \tag{A3}$$

which is a linear constraint.

On the other hand, if A is uncertain, Eq. (A2) is equivalent to the following set of linear constraints:

$$\begin{aligned} &\Gamma \sum_{l=1}^{L} w_i^l + \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 \leq 0 \\ &-b_i^l - \boldsymbol{a}_i^l \boldsymbol{x} \leq w_i^l, \quad \forall\ l \in 1, \ldots, L \\ &b_i^l + \boldsymbol{a}_i^l \boldsymbol{x} \leq w_i^l, \quad \forall\ l \in 1, \ldots, L \end{aligned} \tag{A4}$$

A.2. Ellipsoidal Uncertainty Set

If the perturbation set $\mathcal{Z}$ is an ellipsoidal, i.e.,

$$\sum_{l=1}^{L} \frac{\zeta_l^2}{\sigma_l^2} \leq \Gamma^2$$

then the robust formulation of the ith constraint is equivalent to

$$\Gamma \sqrt{\sum_{l=1}^{L} \sigma_l^2 (-b_i^l - \boldsymbol{a}_i^l \boldsymbol{x})^2} + \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 \leq 0 \tag{A5}$$

which is a second-order conic constraint.

If only b is uncertain (i.e., $\mathbb{A}^l = 0, \forall l = 1, 2, \ldots, L$), then Eq. (A5) becomes

$$\sum_{l=1}^{L} \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 + \Gamma \sqrt{\sum_{l=1}^{L} \sigma_l^2 (b_i^l)^2} \leq 0 \tag{A6}$$

which is a linear constraint.

A.3. Norm-1 Uncertainty Sets

If the perturbation set represented by $\mathcal{Z}$ is a norm-1 uncertainty set (i.e., $\|\zeta\|_1 \leq \Gamma$), then the robust constraint is

$$\sum_{l=1}^{L} \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 + \Gamma \max_{l=1,\ldots,L} |b_i^l| \leq 0 \tag{A7}$$

when $\mathbb{A}^l = 0$, and

$$\begin{aligned} &\Gamma w_i + \boldsymbol{a}_i^0 \boldsymbol{x} + b_i^0 \leq 0 \\ &-b_i^l - \boldsymbol{a}_i^l \boldsymbol{x} \leq w_i, \quad \forall\ l \in 1, \ldots, L \\ &b_i^l + \boldsymbol{a}_i^l \boldsymbol{x} \leq w_i, \quad \forall\ l \in 1, \ldots, L \end{aligned} \tag{A8}$$

if $\mathbb{A}^l \neq 0$. Note that for this type of uncertainty, the robust constraints are linear.

References

[1] Zang, T. A., Hemsch, M. J., Hilburger, M. W., Kenny, S. P., Luckring, J. M., Maghami, P., Padula, S. L., and Stroud, W. J., "Needs and Opportunities for Uncertainty-Based Multidisciplinary Design Methods for Aerospace Vehicles," NASA TM 2002-211462, July 2002, https://www.google.com/books/edition/Needs_and_Opportunities_for_Uncertainty/9aM9AQAAMAAJ?hl=en&gbpv;=1 [retrieved 10 March 2021].

[2] Yao, W., Chen, X., Luo, W., van Tooren, M., and Guo, J., "Review of Uncertainty-Based Multidisciplinary Design Optimization Methods for Aerospace Vehicles," *Progress in Aerospace Sciences*, Vol. 47, No. 6, 2011, pp. 450–479.
https://doi.org/10.1016/j.paerosci.2011.05.001

[3] York, M. A., Öztürk, B., Burnell, E., and Hoburg, W. W., "Efficient Aircraft Multidisciplinary Design Optimization and Sensitivity Analysis via Signomial Programming," *AIAA Journal*, Vol. 56, No. 11, 2018,

pp. 4546–4561.
https://doi.org/10.2514/1.J057020

[4] Gallard, F., Meaux, M., Montagnac, M., and Mohammadi, B., "Aerodynamic Aircraft Design for Mission Performance by Multipoint Optimization," *21st AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2013-2582, 2013, pp. 1–17.
https://doi.org/10.2514/6.2013-2582

[5] Liem, R. P., Kenway, G. K., and Martins, J. R., "Multimission Aircraft Fuel-Burn Minimization via Multipoint Aerostructural Optimization," *AIAA Journal*, Vol. 53, No. 1, 2015, pp. 104–122.
https://doi.org/10.2514/1.J052940

[6] Liem, R. P., Martins, J. R., and Kenway, G. K., "Expected Drag Minimization for Aerodynamic Design Optimization Based on Aircraft Operational Data," *Aerospace Science and Technology*, Vol. 63, April 2017, pp. 344–362.
https://doi.org/10.1016/j.ast.2017.01.006

[7] Diwekar, U., *Introduction to Applied Optimization*, Vol. 22, Springer, New York, 2008, pp. 135–141.
https://doi.org/10.1007/978-0-387-76635-5

[8] Bertsimas, D., Brown, D. B., and Caramanis, C., "Theory and Applications of Robust Optimization," *Society for Industrial and Applied Mathematics*, Vol. 53, No. 3, 2011, pp. 464–501.
https://doi.org/10.1137/080734510

[9] Kall, P., and Stoyan, D., "Solving Stochastic Programming Problems with Recourse Including Error Bounds," *Mathematische Operationsforschung und Statistik. Series Optimization*, Vol. 13, No. 3, 1982, pp. 431–447.
https://doi.org/10.1080/02331938208842805

[10] Higle, J. L., and Sen, S., "Stochastic Decomposition: An Algorithm for Two-Stage Linear Programs with Recourse," *Mathematics of Operations Research*, Vol. 16, No. 3, 1991, pp. 650–669.
https://doi.org/10.1287/moor.16.3.650

[11] Pereira, M., and Pinto, L., "Multi-Stage Stochastic Optimization Applied to Energy Planning," *Mathematical Programming*, Vol. 52, Nos. 1–3, 1991, pp. 359–375.
https://doi.org/10.1007/BF01582895

[12] Chen, X., Sim, M., and Sun, P., "A Robust Optimization Perspective on Stochastic Programming," *Operations Research*, Vol. 55, No. 6, 2007, pp. 1058–1071.
https://doi.org/10.1287/opre.1070.0441

[13] Bertsimas, D., Gupta, V., and Kallus, N., "Data-Driven Robust Optimization," *Mathematical Programming*, Vol. 167, No. 2, 2018, pp. 235–292.
https://doi.org/10.1007/s10107-017-1125-8

[14] Goh, J., and Sim, M., "Distributionally Robust Optimization and Its Tractable Approximations," *Operations Research*, Vol. 58, No. 4, 2010, pp. 902–917.
https://doi.org/10.1287/opre.1090.0795

[15] Bertsimas, D., and Sim, M., "The Price of Robustness," *Operations Research*, Vol. 52, No. 1, 2004, pp. 35–53.
https://doi.org/10.1287/opre.1030.0065

[16] Shmoys, D., and Swamy, C., "Stochastic Optimization is (Almost) as easy as Deterministic Optimization," *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, IEEE, Piscataway, NJ, 2004, pp. 228–237.
https://doi.org/10.1109/focs.2004.62

[17] Duffin, R., Peterson, E., and Zener, C., *Geometric Programming: Theory and Application*, Wiley, New York, 1967, pp. 1–15.
https://doi.org/10.1002/zamm.19670470843

[18] Gondzio, J., "Interior Point Methods 25 Years Later," *European Journal of Operational Research*, Vol. 218, No. 3, 2012, pp. 587–601.
https://doi.org/10.1016/j.ejor.2011.09.017

[19] Boyd, S., Kim, S.-J., Vandenberghe, L., and Hassibi, A., "A Tutorial on Geometric Programming," *Optimization and Engineering*, Vol. 8, No. 1, 2007, pp. 67–127.
https://doi.org/10.1007/s11081-007-9001-7

[20] Hoburg, W. W., "Aircraft Design Optimization as a Geometric Program," Ph.D. Thesis, Univ. of California Berkeley, Berkeley, CA, 2013.

[21] Burton, M., and Hoburg, W., "Solar and Gas Powered Long-Endurance Unmanned Aircraft Sizing via Geometric Programming," *Journal of Aircraft*, Vol. 55, No. 1, 2018, pp. 212–225.
https://doi.org/10.2514/1.C034405

[22] Kirschen, P. G., York, M. A., Öztürk, B., and Hoburg, W. W., "Application of Signomial Programming to Aircraft Design," *Journal of Aircraft*, Vol. 55, No. 3, 2018, pp. 965–987.
https://doi.org/10.2514/1.C034378

[23] Kirschen, P. G., Burnell, E., and Hoburg, W., "Signomial Programming Models for Aircraft Design," *54th AIAA Aerospace Sciences Meeting*, AIAA Paper 2016-2003, 2016.
https://doi.org/10.2514/6.2016-2003

[24] Saab, A., Burnell, E., and Hoburg, W. W., "Robust Designs via Geometric Programming," Preprint, submitted 22 Aug. 2018, https://arxiv.org/abs/1808.07192.

[25] Saab, A., and Öztürk, B., *Robust: A Python Package for Geometric Programming Under Uncertainty*, Software Package, Ver. 0.0.0, 2020, https://robust.readthedocs.io/en/latest/ [retrieved 10 March 2021].

[26] Burnell, E., Damen, N. B., and Hoburg, W., "GPkit: A Human-Centered Approach to Convex Optimization in Engineering Design," *Proceedings of the Computer Human Interface (CHI) Conference on Human Factors in Computing Systems*, Association for Computing Machinery, New York, 2020, pp. 1–13.
https://doi.org/10.1145/3313831.3376412

[27] Öztürk, B., "Conceptual Engineering Design and Optimization Methodologies using Geometric Programming," M.S. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, Feb. 2018.

[28] Lipp, T., and Boyd, S., "Variations and Extension of the Convex–Concave Procedure," *Optimization and Engineering*, Vol. 17, No. 2, 2016, pp. 263–287.
https://doi.org/10.1007/s11081-015-9294-x

[29] Chassein, A., and Goerigk, M., "Robust Geometric Programming is Co-NP Hard," Technische Univ. Kaiserslautern, Kaiserslautern, Germany, 2014, pp. 1–6, http://nbn-resolving.de/urn:nbn:de:hbz:386-kluedo-39380 [retrieved 10 March 2021].

[30] Hsiung, K. L., Kim, S. J., and Boyd, S., "Tractable Approximate Robust Geometric Programming," *Optimization and Engineering*, Vol. 9, No. 2, 2008, pp. 95–118.
https://doi.org/10.1007/s11081-007-9025-z

[31] Tao, T., "Design, Optimization, and Performance of an Adaptable Aircraft Manufacturing Architecture," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 2018.

[32] Hoburg, W., Kirschen, P., and Abbeel, P., "Data Fitting with Geometric-Programming-Compatible Softmax Functions," *Optimization and Engineering*, Vol. 17, No. 4, 2016, pp. 897–918.
https://doi.org/10.1007/s11081-016-9332-3

[33] Drela, M., "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," *Low Reynolds Number Aerodynamics*, edited by Mueller, T. J., Springer, Berlin, 1989, pp. 1–12.
https://doi.org/10.1007/978-3-642-84010-4_1

[34] Ozturk, B., Burton, M. J., and Hoburg, W. W., "Design of an Unmanned Aerial Vehicle for Long-Endurance Communication Support," *AIAA Aviation Forum, 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA Paper 2017-4148, 2017, pp. 1–18.
https://doi.org/10.2514/6.2017-4148.

[35] ApS, M., *MOSEK Optimizer API for Python*, Software Package, Ver. 9.0, 2020, https://docs.mosek.com/9.0/pythonapi/index.html.

[36] Kirschen, P. G., and Hoburg, W. W., "The Power of Log Transformation: A Comparison of Geometric and Signomial Programming with General Nonlinear Programming Techniques for Aircraft Design Optimization," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, AIAA Paper 2018-0655, 2018.
https://doi.org/10.2514/6.2018-0655

R. Ghanem
Associate Editor