Regular Paper

Isoperformance: Analysis and Design of Complex Systems with Desired Outcomes*

Olivier L. de Weck[1, †] and Marshall B. Jones[2]

[1] *Engineering Systems Division, Department of Aeronautics & Astronautics,* 77 *Massachusetts Avenue,* 33-410, *Massachusetts Institute of Technology, Cambridge,* MA 02139

[2] *Behavioral Science Department,* 500 *University Drive, Room* C1728, *Pennsylvania State University, Hershey,* PA 17033

Received 27 October 2004; Accepted 17 October 2005, after one or more revisions
Published online in Wiley InterScience (www.interscience.wiley.com).
DOI 10.1002/sys.20043

ABSTRACT

The design of technical systems such as automobiles and spacecraft has traditionally focused exclusively on performance maximization. Many organizations now realize that such an approach must be balanced against competing objectives of cost, risk, and other criteria. If one is willing to give up some amount of performance relative to the best achievable performance level, one introduces slack into system design. This slack can be invested in creating better outcomes overall. One way to achieve this is to balance the requirements among contributing subsystems such that the number of active constraints is minimized, while still achieving the desired system performance. This paper introduces a methodology called "isoperformance" as a means of identifying and evaluating a performance-invariant set of design solutions, which are efficient in terms of other criteria such as cost, risk, and lifecycle properties. Isoperformance is an inverse design method that starts from a desired vector of performance requirements and works backwards to identify acceptable solutions in the

*This work was partially supported by the NASA Goddard Space Flight Center under Contracts No. NAG5-6079 and No. NAG5-7839 and by the Jet Propulsion Laboratory under Contract No. JPL 961123.

†Author to whom all correspondence should be addressed (e-mail: deweck@mit.edu).

Systems Engineering, Vol. 9, No. 1, 2006

design space. To achieve this, gradient-based contour following is implemented as a multivariable search algorithm that manipulates the null set of the Jacobian matrix. Use of the method is illustrated with two examples from spacecraft design and human performance in sports. © 2006 Wiley Periodicals, Inc. Syst Eng 9: 45–61, 2006

Key words: system design; isoperformance; multiobjective optimization; constraints; gradient-based search; inverse design methods; sensitivity analysis

1. INTRODUCTION

Engineers have been traditionally conditioned to think in particular ways about the challenging task of system design. First, emphasis has been historically placed on performance maximization, which was evident in projects such as Apollo (1962–1973), the development of the Concorde (1956–1976) or even current practices in Formula 1 race car design. Second, the dominant way in which design is taught and practiced is in the mode of “forward analysis.” Choices in the design space such as material selection, geometry specification or controller gain settings are evaluated in terms of their impact on system response and objectives such as range, speed, and more recently fuel consumption. If initial design choices are unsatisfactory, engineers typically iterate on their designs until such quantities have been maximized or minimized, subject to a number of other technical and financial feasibility constraints. This modus operandi is facilitated by the fact that, with the possible exception of buckling and other instability situations, each vector of choices, $\mathbf{x}$, in the design space can usually be uniquely mapped to an expected outcome in the objective space, $\mathbf{x} \mapsto \mathbf{J}(\mathbf{x})$.

This paradigm has its roots in an era where system performance was the prime driver of competitiveness and superiority. This is particularly true for aerospace engineering, which has only recently come to realize that performance is not necessarily limited by physical phenomena (e.g., transonic drag, atmospheric properties in the stratosphere) alone, but that “pushing the performance envelope” too hard can have detrimental consequences for other aspects of the system. We find the following quote by Schrage et al. [1991] to be particularly appropriate: “The experience of the 1960’s has shown that for military aircraft the cost of the final increment of performance usually is excessive in terms of other characteristics and that the overall system must be optimized, not just performance.”

A more natural mode of thought is not to strive for the “best achievable” system performance, but acceptable performance that is “good enough,” depending on contractually specified requirements, the state of competition in various market segments, and the need to achieve desired robust functionality at the lowest possible cost. In that case the desired performance levels become known quantities that can serve as targets for system designers. This is the focus of so-called inverse design methods, whose goal it is to find a set of solutions in the design space that satisfy a set of performance targets in the objective space, $\mathbf{J} \mapsto \mathbf{x}(\mathbf{J})$. The main challenge lies in the nonuniqueness of the problem. Since the number of design variables generally far exceeds the number of objectives, there might be many, often infinitely many design vectors, $\mathbf{x}_i$, that satisfy the vector of performance targets, $\mathbf{J}_{req}$. This, of course, assumes that none of the targets are utopian, i.e., that the targets are feasible based on physics and available technologies (but not necessarily based on available resources).

Isoperformance is a method that addresses this problem by first obtaining a performance invariant set of system design solutions and subsequently reducing these to an efficient set, when evaluated against other criteria such as cost and risk. Said more simply, it is an inverse design method that allows system designers to ask: “What is the set of feasible designs, $\mathbf{X}_{iso}$, that satisfies all performance targets within some tolerance, while optimizing other criteria such as cost and risk?”.

Let us consider a simple example from rocketry to illustrate this point. The primary performance of a rocket is measured by the change in velocity, Δv, that it can achieve for a specified payload mass m_p. The rocket equation by Tsiolkovsky [1903] states that

$$\Delta v = v_f - v_i = g \cdot I_{sp} \cdot \ln\left(\frac{m_i}{m_f}\right), \tag{1}$$

where v_i is the initial velocity in m/s, v_f is the final velocity, g is Earth’s mean gravitational acceleration at the surface, $g = 9.81$ m/s^2, I_{sp} is the specific impulse of the propulsion system in seconds, m_i is the initial mass, and m_f is the final mass after engine cutoff. The initial mass can be written as

$$m_i = m_s + m_f + m_p = (1 + \alpha)m_f + m_p, \tag{2}$$

where m_s is the structural (empty) mass of the rocket, m_f is the mass of fuel, and m_p is the payload mass. For simplicity one can assume that α represents the structural mass of the rocket as a fraction of total fuel mass. Typical values for α are on the order of 0.08–0.12. The final mass—assuming all fuel is consumed—is simply

$$m_f = m_s + m_p = \alpha m_f + m_p. \qquad (3)$$

If we were to design a rocket for launching payloads to low Earth orbit, a typical performance requirement would be $J_{req} = \Delta v_{req} = 9500$ m/s. This requirement allows achieving orbital velocity while absorbing drag, gravity, and flight path turning losses during ascent. In other words, satisfactory performance is defined as accelerating a payload mass of say, $m_p = 5000$ kg, by the velocity increment, Δv_{req}. If the actual performance achieved is below this level, a payload would not reach or remain in orbit and the system would be useless. On the other hand, if the system significantly exceeds the requirement, it will achieve its purpose but be overdesigned and, as a consequence, be heavier, more complex, and more costly than really needed. So, what is the family of feasible rocket designs that satisfies the required performance level?

Isoperformance is a method for thinking through and solving this problem. After specifying the required function and performance level, one must understand which variables in the system the designer can change independently, and which ones are constrained by nature or man to be fixed constants. In the case of rocket design we may choose the I_{sp} (by selecting a particular oxidizer/propellant combination) and size of the rocket (by deciding on the quantity of fuel). Of course, architectural decisions such as the number of rocket stages must be considered as well and we will return to this point later on. Let $\mathbf{x} = [x_1 \quad x_2]^T$ be the design vector with $x_1 = I_{sp}$ and $x_2 = m_f$. The vector of fixed parameters is $p = [p_1 \; p_2 \; p_3]^T$ with $p_1 = g = 9.81$ m/s, $p_2 = m_p = 5000$ kg, and $p_3 = \alpha = 0.1$.

Given the fact that we have $n = 2$ design variables in this example, we can easily evaluate all possible designs between technologically feasible upper and lower bounds:

$$x_{1,LB} = 300 \le x_1 \le x_{1,UB} = 500,$$

$$x_{2,LB} = 5 \cdot 10^4 \le x_2 \le x_{2,UB} = 10^6. \qquad (4)$$

We can discretize this design space into small increments, Δx_i, and evaluate all possible combinations of specific impulse and fuel mass. The result is shown in Figure 1 and depicts the characteristic "wall" caused by the rocket equation. As I_{sp} is decreased by a small amount, it takes more and more propellant to achieve the same Δv to make up for the loss of combustion

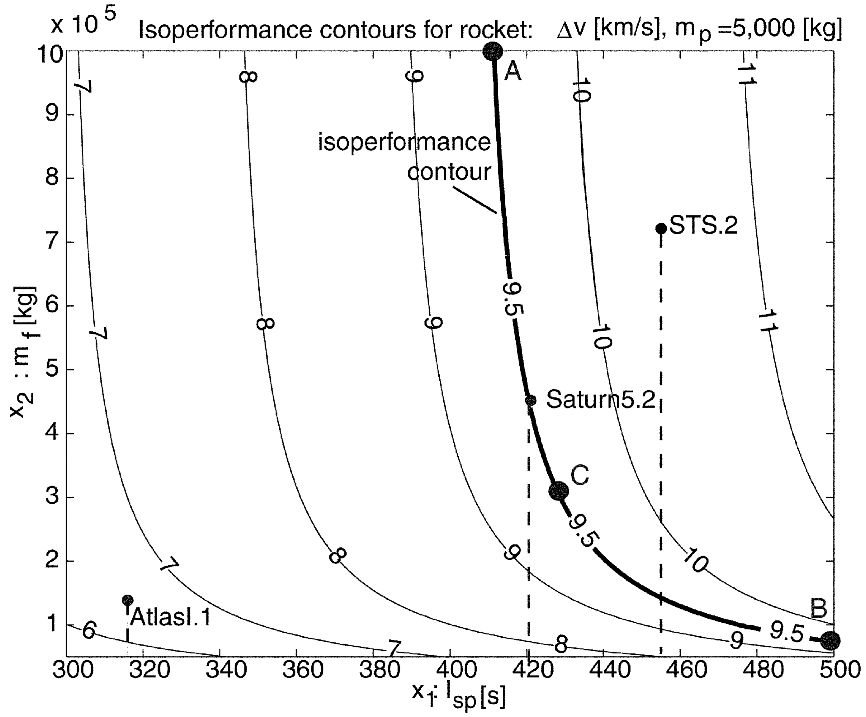


Figure 1. Isoperformance contours of v [km/s] for simple rocket example.

efficiency. The plot shows the isoperformance contour of interest (9.5 km/s) and the relationship that must be maintained between the design variables x_1 and x_2 to stay on the contour. Thus, there are an infinite number of choices between the two extreme designs: A and B. A uses propulsion technology which is easily achievable, while B assumes an I_{sp} of 500 s, which is very difficult to achieve even with state-of-the-art liquid-hydrogen, liquid-oxygen propulsion technology. Said more plainly, design A makes the job of the propulsion subsystem designers easy but requires a very large rocket which creates structural and controls challenges. Design B, on the other hand, is small and compact, but very challenging from a propulsion technology standpoint. It is interesting to note where various existing rocket stage designs fall in that space. The second stage of the Saturn V launch vehicle (Saturn5.2) falls exactly onto the isoperformance contour and represents a compromise design (close to design C) that meets performance requirements but balances the difficulty between structural and propulsion subsystems.[1] Other systems such as the Space Shuttle second stage (STS.2), would be significantly overdesigned or underdesigned (Atlas-I first stage), respectively.

This example, while accurate based on first principles, does not capture some of the most important considerations in selecting among designs A, B, or C. One example of such a secondary criterion is the time required to prepare the vehicle for launch. While operating along the isoperformance contour (Fig. 1) may be efficient from a propulsion standpoint it implies that cryogenic fluids will be used (LH_2, LOX) and that a rocket cannot be readied for launch with only a few minutes notice. This is the main reason why designers of ICBMs chose lower I_{sp} designs that could be readied for launch in minutes rather than hours. Accepting lower propulsion performance was acceptable in that case because payloads were smaller and orbital velocities did not have to be achieved due to ballistic reentry.

Finding the isoperformance contour in this example was trivial because we only considered two design variables and a single objective. What if multiple objectives were to be considered together and the vector of design variables were significantly larger? In that case finding the set of isoperformance solutions is not trivial, and we must find ways to systematically search the design space for acceptable solutions. And, once such solutions are obtained, we must find means to reduce the large set of alternatives to a smaller set that can be presented to decision-makers. Finally, criteria for selecting a particular solution in the reduced set must be discussed. These three sequential questions give rise to the three steps of the isoperformance methodology presented in this paper.

2. LITERATURE REVIEW AND PROBLEM FORMULATION

2.1. Related Literature

A system is understood broadly as a complex set of human and/or artificial components that interact to achieve a desired function. Performance is a quantitative measure of how well this function is executed. Four of the main tensions during system or product development have been identified by Maier and Rechtin [2000] (see Fig. 2). One of the important tasks of a systems engineer or program manager is to identify, quantify, and resolve these tensions. An increase in system performance can generally only be achieved by increasing cost, stretching project schedules, accepting a higher level of risk, or a combination of these according to Shishko et al. [1995].

Isoperformance is an operational method that builds upon Herbert Simon's [1996] notion of "satisficing." Satisficing is to "accept a choice or judgment as one that is good enough, one that satisfies." According to Herb Simon, who coined the term, the tendency to satisfice shows up in many cognitive tasks such as playing games, designing under time and schedule pressure, and making financial decisions where people typically do not or cannot search for the optimal solution.

The idea of holding a performance metric or value of an objective function constant and finding the corresponding "contours" has been previously explored by researchers in other areas. Gilheany [1989], for example, presented a methodology for optimally selecting dampers for multi-degree-of-freedom systems. In that particular work the contours of equal values of the objective function are found as a function of the damping coefficients. The term "isoperformance" was origi-

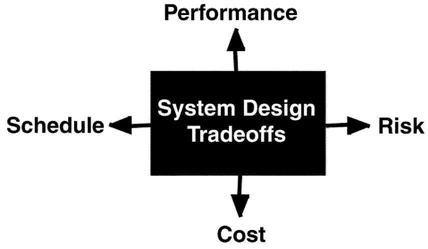


Figure 2. Tensions during systems architecting and design [Maier and Rechtin, 2000: 83].

[1] Note that while the position of the existing rockets in Figure 1 is correct, the actual Δv and payload masses associated with them are different from the ones assumed in this simple example.

nally coined in the area of human factors research. Work has been done by Kennedy, Jones, and coworkers [Kennedy, Turnage, and Jones, 1990; Jones and Kennedy, 1996] on the need within the U.S. Department of Defense to improve systems performance through better integration of men and women into military systems (human factors engineering). They present the application of isoperformance analysis in military and aerospace systems design, by trading off equipment, training variables, and user characteristics. Once the level of operational performance in these systems is settled upon (e.g., "pilot will check out all aircraft systems within 30 seconds"), tradeoffs among equipment variables, adaptation, training, and individual predisposing factors can be made. Jones in particular has also linked isoperformance to personnel selection [2000].

Inverse design has recently become a topic in aerodynamics where one would like to automatically generate airfoil geometry (usually as a sequence of *x*/*y* coordinates of control points) for a prescribed pressure distribution over a wing (see, for example, Kim, Kim, and Rho [1999]). Level Set Methods in mathematics also have the potential to represent isoperformance surfaces (see Osher and Fedkiw [2002]). So far, however, these advanced geometrical techniques have been mainly applied to visualization and computer graphics.

A number of researchers such as Taguchi, Cook [1997], and Messac [1996] have recognized that system requirements typically fall into one of three classes: "smaller-is-better" (SIB), "larger-is-better" (LIB), and "nominal-is-better" (NIB) (see Fig. 3). In automotive design, for example, a target average fuel economy [mpg] might have to be achieved (NIB) for a vehicle, while at the same time vehicle variable manufacturing cost [$] might have to be minimized (SIB) with interior roominess [m^3] being maximized (LIB). Which of these objectives are considered as NIB target values and which ones can be maximized or minimized depends on the particular circumstances. Typically, however, such objectives are counteracting. Large interior volume would tend to increase drag, which in turn decreases fuel economy. A target vehicle fuel economy can be achieved by trading off fuel capacity, empty weight, and engine displacement among other variables. The isoperformance approach assumes that desired performance targets J_{req} are known, i.e., that the key performance objectives are captured as NIB and that they must be achieved first. Traditional performance maximization always assumes (LIB) at the expense of the other objectives. Isoperformance, on the other hand, fixes the amount of performance at an acceptable level (NIB) and trades off the other objectives with respect to each other.

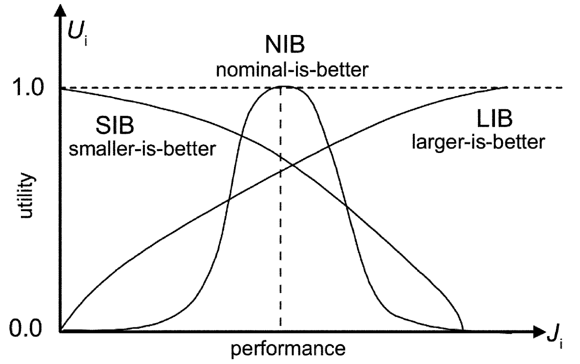


Figure 3. Normalized utility curves u_i of the *i*th system objective J_i represented by a monotonically decreasing (SIB), increasing (LIB), or concave function (NIB).

This is similar in philosophy to Physical Programming, developed by Messac [1996] with the difference that in Physical Programming all objectives, LIB, NIB, and SIB, are mapped onto a unitless scale of goodness and combined together into an overall system utility. It is this unitless measure that is then optimized. The result of Physical Programming is a single point design that will maximize overall utility depending on the breakpoints that designers select for various levels of desirability. For example, for system mass the user might select the following ranges: Highly Desirable < 250 (kg), Desirable 250–275, Tolerable 275–300, Undesirable 300–325, Highly Undesirable 325–350, Unacceptable > 350. While this approach is intuitive and ensures that all objectives contribute somewhat to overall system utility, there is no guarantee that those performance objectives that are deemed critical will indeed be met. Of course, additional points can be obtained in Physical Programming by rerunning the optimization for different settings of the desirable-undesirable range settings and different weightings.

In isoperformance on the other hand we first extract the subset of solutions that strictly satisfy the NIB requirements. We carefully analyze and visualize this set of solutions and try to extract engineering insight from them. A second step is to evaluate the performance invariant solutions in terms of their SIB and LIB objectives and to filter the set to only those solutions that are nondominated according to the SIB and LIB objectives. The third step is to select one (or more) designs from this Pareto [1906] set for further consideration.

Unfortunately, most inverse design problems addressed in the literature are of small scale and don't address the curse of dimensionality of complex system design. What is needed is a more general methodology that can deal with system design problems on a larger

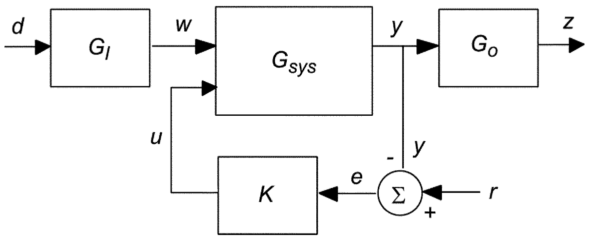


Figure 4. Block diagram of dynamic system with feedback control.

scale that have both a large number of design variables, objectives, constraints and potentially both artificial and human subsystems and components. The next sections will present the formulation and solution of the isoperformance method as one step in this direction. The method will be contrasted with a more traditional all-in-one multiobjective optimization formulation and two examples will be presented, one from spacecraft design, the other from human performance in sports.

2.2. Notation

Let a general dynamic system be represented by a block diagram (Fig. 4) whereby the system is subject to both exogenous inputs d, which enter the system as filtered disturbances w, as well as control inputs u. The system responds by a measurable system output, y. A system controller, K, may be present in an attempt to minimize the error signal, e, between the desired system behavior, r (reference signal) and the measured response. The performance of the system, z, may be different from the measured output, y. The system objectives are typically functions of the performance output signal $z(t)$ such that $J = f(z)$.

Many systems may be described in this framework; even if the exact form the transfer functions G_I, G_{sys}, G_O, and K may not always be known. Table I maps the various quantities to the systems of interest in this paper. This list is not meant to be exhaustive, but rather illustrative. As the states of dynamical systems evolve over time, their behavior can be measured and influenced to the extent allowed by the system design.

If the system is assumed to be linear and time-invariant, it may be described by a set of governing equations such as

$$\dot{q} = A_{sys}(x_i, p_j)q + B_{sys}(x_i, p_j)d,$$

$$z = C_{sys}(x_i, p_j)q + D_{sys}(x_i, p_j)d, \quad \text{where } i = 1, 2, \ldots, n,$$

$$G_{sys} = C_{sys}[sI - A_{sys}]^{-1}B_{sys} + D_{sys}. \tag{5}$$

This may not always be a good assumption, but we will follow this framework for now, because it is allows us to rigorously define the relationship between design variables, fixed parameters, and system performance. Moreover, the satellite in the first example is modeled as a dynamic, linear time-invariant system.

The dynamic states of the system, q, are generally internal, and the matrices describing the system properties, A, B, C, and D clearly depend on how the system has been designed. The expression $A_{sys}(x_i, p_j)$ expresses that the A-matrix of the system depends on both settings of the design variables x_i where $i = 1, 2, \ldots n$, as well as the fixed parameters p_j. A performance objective for the system may be defined as an average, maximum, minimum, cumulative, root-mean-square, or other statistical measure of system performance, $z(t)$. For example, if $z(t)$ is a time-varying performance output signal,

Table I. Examples of Input, Output, and Performance Variables for Various Systems

System	Exogenous Input d	Measured Signal y	Control Action u	System Performance z	System Objective J
Rocket	atmospheric density, temperature	engine mass flow rate	throttle[§§] opening, fin angle	altitude, velocity	total Δv achieved
Car	road profile, surface roughness	speed, engine RPM	throttle position, gear, wheel angle	speed, torque	average fuel economy, lap time
Scientific Satellite	solar pressure	image centroid position	reaction wheel speed	line-of-sight angles	root-mean-square pointing accuracy
Baseball Team	umpire, fan behaviour, injuries	ERA, RBI etc. statistics	buy, sell, activate, task players	% games won	final standing at season end

[§§] Only liquid fuel rockets can typically be throttled.

then the objective may be to minimize the root-mean-square (RMS) value of that signal.

$$J_z = f(z),$$

$$\text{e.g., } J_z = \|z\|_2 = \mathrm{E}[z^T z]^{1/2} = \left(\frac{1}{T}\int_0^T z(t)^2 dt\right)^{1/2} \quad \text{RMS.} \tag{6}$$

An isoperformance requirement can then be formulated as

$$J_z(x_{iso,i}) \equiv J_{z,req} \qquad \forall \ i = 1, \ldots, n, \tag{7}$$

whereby a two-side tolerance band, τ, may be allowed for practical and numerical reasons:

$$\tau: \ \left|\frac{J_z(x_{iso,i}) - J_{z,req}}{J_{z,req}}\right| \leq \tau. \tag{8}$$

Thus, $J_{z,req}$ represents not an ideal but acceptable level of RMS performance of signal $z(t)$. In the first example, $z(t)$ will represent the residual line-of-sight (LOS) jitter of a space telescope. Attempting to drive this requirement to zero or a number that is significantly below the resolution of the on-board imaging instruments would be difficult to achieve and wasteful.

The isoperformance problem is succinctly stated in the following section.

2.3. Isoperformance Problem Formulation

The isoperformance problem is to find a set of design vectors, $x^k{}_{iso}$, that are in the isoperformance set such that the isoperformance constraints, feasibility constraints and side bounds are all satisfied.

Step 1—Find performance-invariant set

Find all $\mathbf{x}_{iso}^k \subseteq \mathbf{X}_{iso}$

such that for an objective vector $\mathbf{J}_z = [J_{z,1} \cdots J_{z,m}]^T$ the following constraints are satisfied:

$$\left|\frac{\mathbf{J}_z(\mathbf{x}_{iso}^k, \mathbf{p}) - \mathbf{J}_{z,req}}{\mathbf{J}_{z,req}}\right| \leq \tau \quad \text{Isoperformance constraints,} \tag{9}$$

$$\mathbf{g}(\mathbf{x}_{iso}^k, \mathbf{p}) \leq 0, \ \mathbf{h}(\mathbf{x}_{iso}^k, \mathbf{p}) = 0 \quad \text{Feasibility constraints,} \tag{10}$$

$$x_{i,LB} \leq x_{iso,i}^k \leq x_{i,UB} \quad \forall \ i = 1, 2, \ldots, n \quad \text{Side bounds.} \tag{11}$$

Recall our rocket design example from Section 1 and the results in Figure 1. We see that the isoperformance set is described by the contour labeled as 9.5 km/s. In this simple case the contour is fully described by the implicit equation

$$\mathbf{X}_{iso}: \ 9.81 \cdot x_1 \cdot \ln\left(\frac{1.1x_2 + 5000}{0.1x_2 + 5000}\right) = 9500. \tag{12}$$

We extract from it a subset of three discrete isoperformance solutions (A, B, and C), $k = 1, 2, 3$:

$$\mathbf{x}_{iso}^1 = \begin{bmatrix} 411 \\ 10^6 \end{bmatrix}, \ \mathbf{x}_{iso}^2 = \begin{bmatrix} 429 \\ 3 \cdot 10^5 \end{bmatrix}, \ \mathbf{x}_{iso}^3 = \begin{bmatrix} 500 \\ 7 \cdot 10^4 \end{bmatrix} \tag{13}$$

These satisfy the side bounds [Eq. (4)]. Note that feasibility constraints were not present in this simple example. In cases where $n >> 2$, there will potentially be many isoperformance solutions, and we can apply secondary criteria to filter these. We assert that these secondary criteria should primarily be associated with cost and risk (Fig. 2) objectives, rather then required system performance.

Step 2—Find efficient subset

Find all $\mathbf{x}_{iso}^{l*} \in \mathbf{X}_{iso}^* \subseteq \mathbf{X}_{iso} \subseteq \mathbb{R}^n$

such that for a vector of secondary (cost and risk) objectives $\mathbf{J}_{cr} = [J_{cr,1} \cdots J_{cr,r}]^T$ there exists no other feasible design vector $\mathbf{x} \in \mathbf{X}_{iso} \subseteq \mathbb{R}^n$ such that:

$$\mathbf{J}_{cr}(\mathbf{x}) \leq \mathbf{J}_{cr}(\mathbf{x}_{iso}^*) \qquad \text{Non-inferiority constraint} \tag{14}$$

$$J_{cr,j}(\mathbf{x}) \leq J_{cr,j}(\mathbf{x}_{iso}^*) \quad \text{Component-wise non-inferiority constraint} \tag{15}$$

for all $j = 1, \ldots, r$ with strict inequality holding for at least one j. Note that the feasibility constraints and side bounds on $\mathbf{x}_{iso}^*$ are already satisfied by virtue of step 1.

In our rocket example the risk of a design might relate to technology maturity of the propulsion system, $J_{cr,1} = (x_1 - I'_{sp})/I'_{sp}$, where $I'_{sp} = 450$ is a value readily achievable by the current state of the art. The cost, on the other hand, might be captured by the size of the vehicle, $J_{cr,2} = \log[\beta x_2]^2$. By setting $\beta = 2 \cdot 10^{-4}$ we normalize the cost of the smallest rocket to unity. We can evaluate the three isoperformance designs, A, B, and C, in terms of $\mathbf{J}_{cr}$ and obtain

$$\mathbf{J}_{cr}(\mathbf{x}^1_{iso}) = \begin{bmatrix} -0.09 \\ 5.3 \end{bmatrix}, \quad \mathbf{J}_{cr}(\mathbf{x}^2_{iso}) = \begin{bmatrix} -0.05 \\ 3.2 \end{bmatrix}, \quad \mathbf{J}_{cr}(\mathbf{x}^3_{iso}) = \begin{bmatrix} .11 \\ 1.3 \end{bmatrix}. \tag{16}$$

All three solutions are noninferior according to Eq. (14), whereby the first solution is the low risk, high cost design, the second is a compromise between manufacturing cost and development risk. and the third is the high risk, low cost solution. Ultimately, large technological risk might translate into high development costs as well.

Step 3—Select final design

Select $\mathbf{x}^{**}_{iso} \in \mathbf{X}^{*}_{iso} \subseteq \mathbf{X}_{iso} \subseteq \mathbb{R}^n$,

according to non-quantified objectives/criteria and stakeholder consensus. (17)

3. ISOPERFORMANCE METHOD AND ALGORITHM

3.1. Advantages over One-Step Multiobjective Optimization

It may be possible to arrive at a similar or even the same solution by collapsing steps 1 and 2 into a single multiobjective optimization problem with two-sided inequality constraints on performance:

$$\min \mathbf{J}_{cr}(\mathbf{x}, \mathbf{p}) = [J_{cr,1} \cdots J_{cr,r}]^T,$$

$$\text{s.t.} \left| \frac{\mathbf{J}_z(\mathbf{x}, \mathbf{p}) - \mathbf{J}_{z,req}}{\mathbf{J}_{z,req}} \right| \le \tau,$$

$$\mathbf{g}(\mathbf{x}, \mathbf{p}) \le 0, \quad \mathbf{h}(\mathbf{x}, \mathbf{p}) = 0,$$

$$x_{i,LB} \le x_i \le x_{i,UB} \quad \forall \; i = 1, 2, \ldots, n. \tag{18}$$

For simple problems the results of the two approaches should be equivalent. For complex system design problems, however, we claim two advantages of the isoperformance method [Eqs. (9)–(16)] over the all-in-one optimization [Eq. (18)]:

1. Engineering insight is obtained by solving the problem in subsequent steps and inspecting and visualizing intermediate results, rather than in an all-in-one optimization.
2. As the tolerance is decreased, $\tau \rightarrow 0$, the isoperformance constraint becomes an inequality constraint which has previously been shown by Kim and de Weck [2005] and others to be difficult to solve using standard multiobjective techniques.

3.2. Isoperformance Method

We can visualize the isoperformance procedure (Fig. 5) by considering the various sets and domains involved in problem solving. Figure 5(a) shows the sets involved. The outer square represents the n-dimensional Euclidian plane, $\mathbb{R}^n$. Designs that we could implemented are bounded by set **B** [side constraints, Eq. (11)] and its subset **F** [feasibility constraints, Eq. (10)]. Furthermore, in step 1 we look for the subset that satisfies the isoperformance constraints, **I**, according to Eq. (9). Among this subset we look for intersection with the Pareto-set **P** [see Eq. (14)], which contains the noninferior solutions from a cost/risk perspective (step 2). The final design $\mathbf{x}^{**}_{iso}$ is said to be from among the efficient set **E**.

Figures 5(b–d) show the various domains that are involved. First, the vector of system objectives is specified in the performance objective space $J_{z,req}$ as a target [Fig. 5(c)]. For a new system about which little is know it is conceivable that the initial target vector is set outside the feasible domain. In that case the best possible outcome is to minimize the Euclidian distance between the target vector and the closest achievable point in the feasible space (goal programming). If the target vector is set inside the feasible domain, the solution will generally be non-unique when the number of design variables exceeds the number of performance objectives, i.e., $n > m$. In step 1 search algorithms are launched from the objective space to identify feasible members of the isoperformance set $\mathbf{X}_{iso}$ in the design space [Fig. 5(b)].[2] Once these have been found, they are evaluated against secondary cost and risk criteria [Fig. 5(d)] and the noninferior set **E** is retained. The final selection of $\mathbf{x}^{**}_{iso}$ is done based on nonquantifiable criteria and stakeholder consensus.

3.3. Gradient-Based Contour Following

There are a number of algorithms that have been proposed to identify the isoperformance set **I**. In de Weck and Miller [2002], the following three algorithms have been developed:

- Branch-and-Bound Design Space Exploration
- Gradient-based Contour Following
- Progressive Vector Spline Approximation

The interested reader is referred to de Weck and Miller [2002] for details on these algorithms. Nevertheless, the second algorithm, gradient-based contour following,

[2] One such algorithm, gradient-based contour following is discussed below.

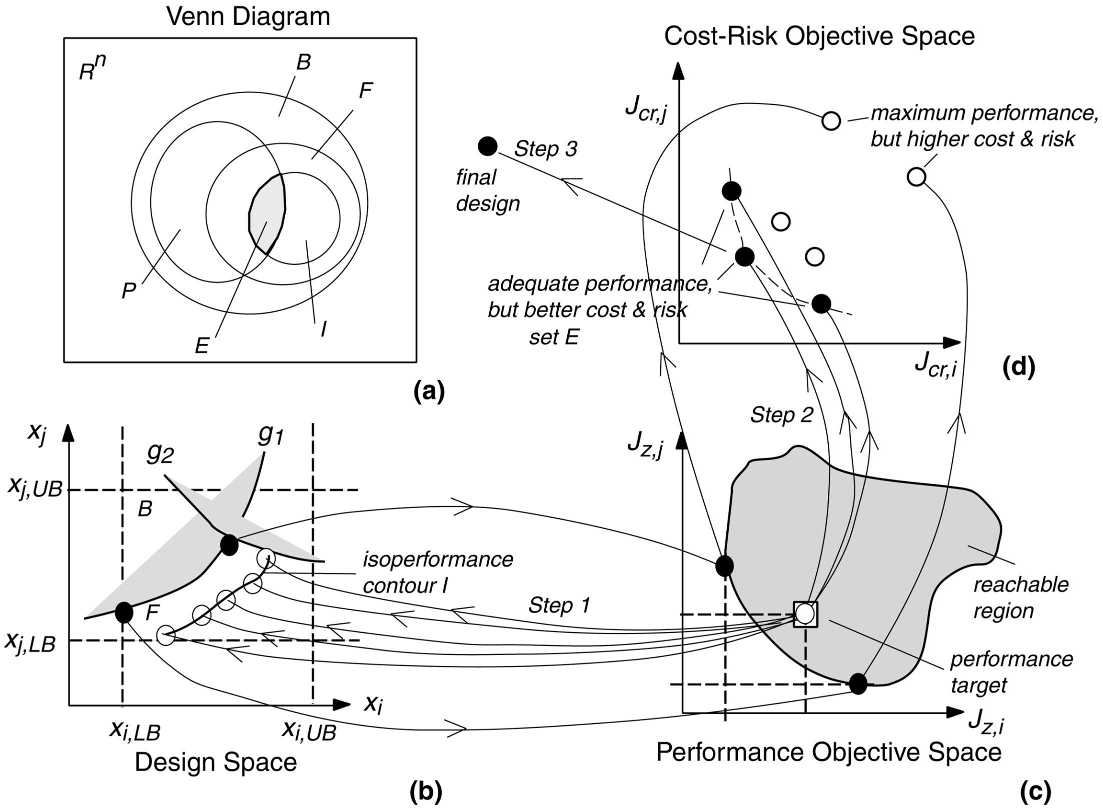


Figure 5. (a) Venn diagram, (b–d) domains mapped in the isoperformance method.

will be briefly discussed here, since it illuminates an interesting mathematical property of the isoperformance approach.

Assume that we start searching for isoperformance solutions from within the feasible domain **F** with a starting point $\mathbf{x}^O$. Furthermore, let

$$J_{z,j}(\mathbf{x}^k+\Delta\mathbf{x}) = J_{z,j}(\mathbf{x}^k) + \underbrace{\nabla J_{z,j}^T\Big|_{\mathbf{x}^k}\Delta\mathbf{x}}_{\text{first-order term}} + \underbrace{\frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}_j\Big|_{\mathbf{x}^k}\Delta\mathbf{x}}_{\text{second-order term}} + \text{H.O.T.} \tag{19}$$

be the Taylor decomposition of the jth objective function evaluated at $\mathbf{x}^k$, where

$$\nabla J_{z,j} = \begin{bmatrix} \frac{\partial J_{z,j}}{\partial x_1} & \cdots & \frac{\partial J_{z,j}}{\partial x_n} \end{bmatrix}^T \tag{20}$$

is the gradient vector of the jth objective function with respect to the design variable vector. Initially, we follow the steepest gradient vector to ensure that we move from the initial point $\mathbf{x}^O$ to a point $\mathbf{x}^k_{iso}$ that satisfies the isoperformance constraint. Once such a point has been found, the algorithm switches to "contour following" mode. Let $\mathbf{x}^k = \mathbf{x}^k_{iso}$ be an isoperformance point that satisfies Eqs. (9)–(11). To find additional isoperformance points, the condition

$$\nabla J_{z,j}^T\big|_{\mathbf{x}^k}\Delta\mathbf{x} \equiv 0 \tag{21}$$

must be satisfied. In other words, we want to find tangential directions that are performance invariant. Since this must be true for all performance objectives, $j = 1, 2, \ldots, m$ at once we define

$$\nabla\mathbf{J}_z = \begin{bmatrix} \frac{\partial J_{z,1}}{\partial x_1} & & \frac{\partial J_{z,m}}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial J_{z,1}}{\partial x_n} & & \frac{\partial J_{z,m}}{\partial x_n} \end{bmatrix} \tag{22}$$

to be the system Jacobian. This is the matrix of partial derivatives of the m performance objectives with respect to the n design variables. Performance invariant directions can be found by singular decomposition of the Jacobian:

$$\mathbf{U}\,\Sigma\,\mathbf{V}^T = \nabla\mathbf{J}_z^T \tag{23}$$

The Σ matrix contains the singular values. Here we are interested in the zero singular values. The corresponding columns of the **V** matrix span the nullspace of the Jacobian.

$$\begin{aligned} \mathbf{U} &= \underbrace{\left[u_1 \ \cdots \ u_m \right]}_{mxm}, \\ \Sigma &= \underbrace{\left[\operatorname{diag}\left(\sigma_1 \ \cdots \ \sigma_m\right) \ 0_{mx(n-m)} \right]}_{mxn}, \\ V &= \left[\begin{array}{c} \underbrace{v_1 \ \cdots \ v_m}_{\text{column space}} \underbrace{v_{m+1} \ \cdots \ v_n}_{\text{nullspace } \mathbf{V}o} \end{array} \right]. \end{aligned} \tag{24}$$

Any linear combination of these n - m null vectors points in a performance invariant direction,

$$\Delta \mathbf{x} = \alpha \cdot (\beta_1 v_{m+1} + \ldots + \beta_{n-m} v_n) = \alpha \mathbf{V}_o \bar{\beta}, \tag{25}$$

where α is an arbitrary, but small, step size and $\bar{\beta}$ is a vector of linear combination parameters. Hence, isoperformance solutions can be obtained from the nullspace of the system Jacobian matrix. Figure 6 shows for a sample problem (de Weck and Miller, [2002]) in three dimensions how the algorithm finds isoperformance surfaces by following performance-invariant directions in the nullspace.

In this paper we are particularly interested in contours and isoperformance surfaces that arise, when the vector $\mathbf{J}_z$ represents the performance of a complex technical or human system. Thus, J_z could represent the pointing performance of a space telescope, average fuel economy of a car, total output of an electrical power grid, or the aptitude of humans as measured by some objective criterion. In economics, relationships of this type are usually called indifference curves. In sensory psychology and physiology, they are often called isofrequency, isochronal or isoelectric curves or contours. These terms all share the prefix iso-, which means "the same." Graphically showing isoperformance results for $n > 3$ is challenging.

3.4. Physics-Based versus Empirical Models

From the discussion thus far it is clear that isoperformance requires some mathematical model that relates the design vector **x** to both performance as well as cost/risk objectives, $\mathbf{J}_z$ and $\mathbf{J}_{cr}$, respectively. The model can be derived from first principles [as was done in Eq. (1)], or obtained from empirical experimental or field observation. In the first case the model is developed and implemented directly (deterministic case). In the second case an empirical model (with embedded statistical uncertainty) must be developed first. This requires a data set that relates experimental factors (**x**) to observed outcomes (**J**). A variety of meta-modeling techniques such as response surfaces, kriging, or neural networks are available for this purpose, and there is an abundant literature on this subject. Figure 7 illustrates these two cases. Note that the isoperformance algorithms used to extract the performance-invariant set $\mathbf{X}_{iso}$, are the same in both cases.

The following two sections give examples of isoperformance applications for both the deterministic case (spacecraft design) as well as the stochastic case (team sports).

4. EXAMPLE 1: SPACECRAFT DESIGN

The design of precision optomechanical systems for remote sensing in space is challenging since it combines tightly coupled disciplines such as structures, optics and controls. When applied to space telescopes such as the "Nexus" spacecraft concept shown in Figure 8, we need

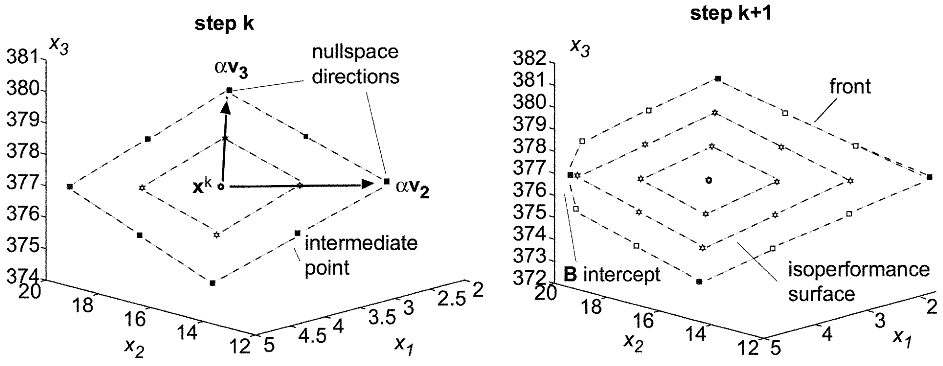


Figure 6. Gradient-based Contour-Following algorithm (n = 3, m = 1).

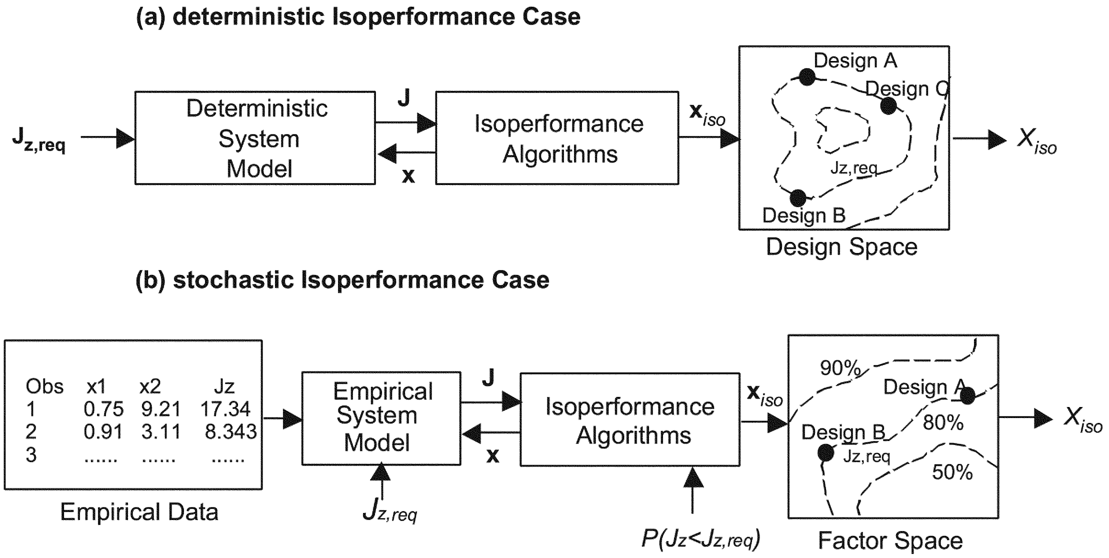


Figure 7. Deterministic versus empirical modeling approaches.

to ensure adequate optical performance, despite the presence of various mechanical and electronic disturbance sources. The system performance is captured by two criteria, $J_{z,1}$, which measures the root-mean-mean-square wave front error (RMMS WFE) and $J_{z,2}$, which measures the root-sum-square (RSS) line-of-sight (LOS) image excursions on the focal plane. Both of these can be simulated using a computer model. Details of the model are presented in de Weck and Miller [2002]. The simulation results for $J_{z,2}$ are for an initial design $\mathbf{x}^O$ and are shown in Figure 9.

The required pointing performance level, $J_{z,2,req} = 5$ [μm], is driven by the size of a pixel on the focal plane (25 μm) so that images will not blur during long exposures. The time trace in Figure 9 represents the motion of the image centroid across the focal plane during 5 s of simulated operations. If the motion is too large, collected photons will be spread over many pixels, and image quality will deteriorate. Setting $J_{z,2,req}$ to a tighter requirement than 5 μm would not yield any benefit to the system as a whole since the performance is inherently limited by pixel size. Enforcing a tighter requirement or even demanding a "zero jitter" system would be nonsensical and expensive. We see that initially

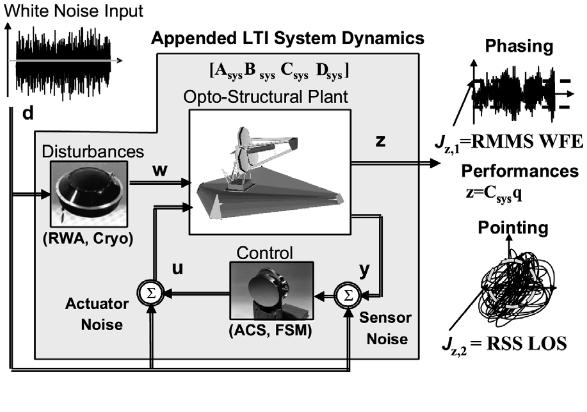


Figure 8. Simulation block diagram of Nexus spacecraft model.

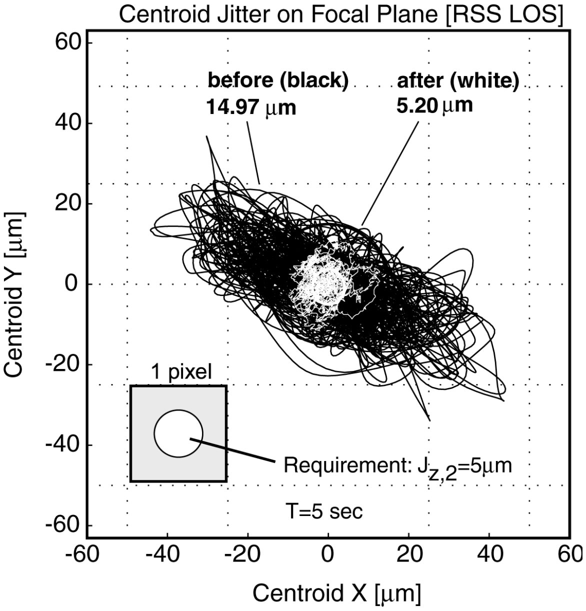
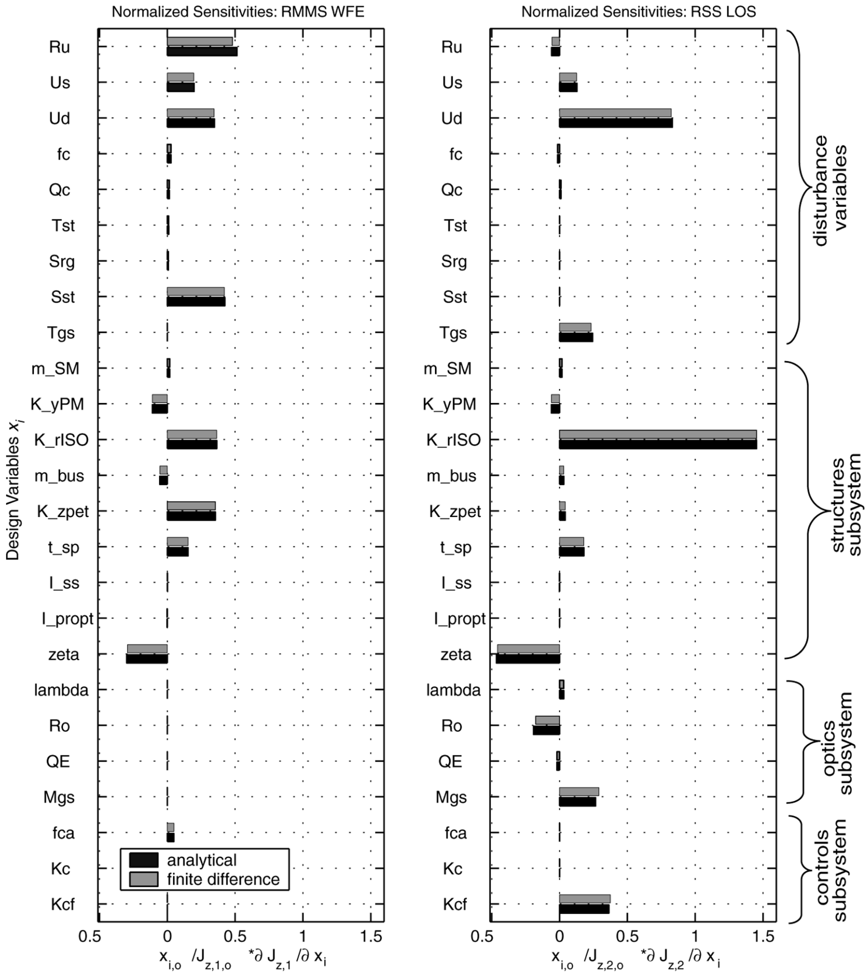


Figure 9. Pointing performance simulation for Nexus space telescope system ($T = 5$ s simulation). Large black trace: initial design $\mathbf{x}^O$; small white trace: isoperformance design $\mathbf{x}^A_{iso}$.


Normalized Sensitivities: RMMS WFE

Normalized Sensitivities: RSS LOS

Figure 10. Bar chart of normalized Jacobian matrix $\nabla \mathbf{J}_z(\mathbf{x}^O)$ for space telescope design.

system performance requirements are not met [$J_{z,2}(\mathbf{x}^O) = 14.97$ µm]. An isoperformance analysis is performed on the system using the procedure described in Sections 3.2 and 3.3.[3]

The performance of the system is a function of many design variables such as the ones shown in the Figure 10. This plot is a graphical bar chart representation of the normalized Jacobian matrix of the system [Eq. (22)] evaluated at the initial design point $\mathbf{x}^O$. The 25 design variables x_i, $i = 1, 2, \ldots, n$, are grouped into disturbance variables and structural, optical, and controls subsystem variables. It is the *combination* of these settings that enables satisfactory system performance. While the details of the model are available elsewhere [de Weck and Miller, 2002], we can see that performance is most sensitive to a subset of these variables (Table II). Isoperformance analysis was performed on this problem and several hundred potential designs that would all meet the performance requirement were identified. How to choose among these?

We introduce three secondary cost-risk objectives that are used to screen solutions from the isoperformance set x_{iso}:

- $$J_{cr,1} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{x_i - (x_{i,LB} + x_{i,UB})/2}{(x_{i,LB} + x_{i,UB})/2}\right|$$

  = closeness to mid-range of design variables,

- $$J_{cr,2} = \left|K_{cf}\right| \quad = \text{magnitude of control gain,} \tag{26}$$

- $$J_{cr,3} = \sum_{i=1}^{n}\left|\frac{\partial J_z}{\partial x_i}\delta x_i\right|$$

  = sensitivity of performance to perturbations.

[3] The other performance metric is simultaneously driven to a target value of $J_{z,1,req} = 20$ µm.

Table II. Subset of the 10 Most Sensitive Design Variables (for Nexus Spacecraft)

subsystem	x_i	initial value	description	units
disturbance	R_u	3000	upper reaction wheel speed	rpm
disturbance	U_s	1.8	static reaction wheel imbalance	g cm
disturbance	U_d	60	dynamic wheel imbalance	g cm^2
disturbance	Q_c	0.005	cryocooler attenuation factor	-
structures	$K_{r,iso}$	3000	isolator joint stiffness	Nm/rad
structures	$K_{z,pet}$	9E+7	primary mirror hinge stiffness	N/m
structures	t_{sp}	0.003	secondary tower wall thickness	mm
optics	T_{gs}	0.040	guide star sensor integration time	sec
optics	M_{gs}	15	maximum guide star magnitude	mag
controls	K_{cf}	2000	fast steering mirror control gain	-

Minimizing the first of these objectives leads to a design where, on average, most design variables will be close to the midpoint between their upper and lower bounds. Since extreme design variable values are often difficult or expensive to achieve, this metric achieves the "best balance" between the various subsystems contributing to overall performance (*A*). The second metric is geared towards minimizing the amount of control gain used. We might call this the "minimum energy" design solution (*B*), while the third metric captures the isoperformance design that is least sensitive to uncertainties in the settings of individual design variables (*C*). We might call this the "most robust" solution. Figure 11 shows a polar plot of these three isoperformance designs (*A*, *B*, *C*), where the actual settings of design variables $x^k_{i,iso}$ are shown in a normalized fashion.

Significant systems engineering insight can be derived from this representation. Design *A* allows almost all variables to be set close to a comfortable midrange, with the exception of M_{gs} (faint stars allowed) and R_u (large wheel speed allowed), whose requirements can be relaxed. Design *B* allows for a small control gain K_{cf}, but in order to still achieve performance, the disturbances sources must be benign (e.g., small imbalances), the reaction wheel isolator must be soft ($K_{r,iso}$ small) and the guide star signal strong (M_{gs} small value = large brightness). Finally, Design *C* is the least uncertain, and we can readily see that this is achieved by strongly suppressing the design variables related to the disturbance sources (U_s, U_d, R_u) and building a stiff, massive structural subsystem ($K_{r,iso}$, t_{sp} are large). The perform-

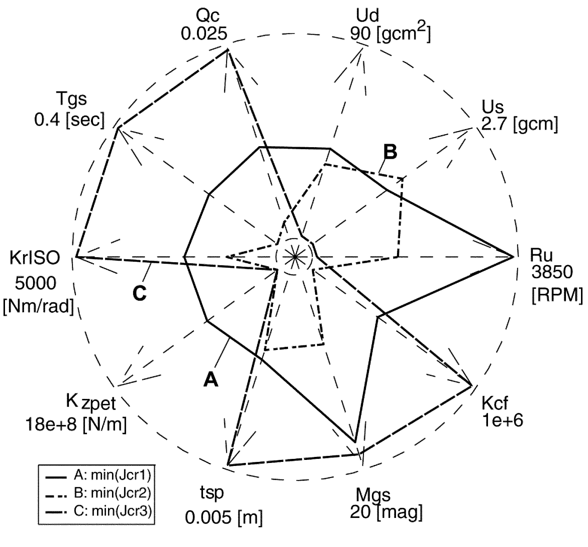


Figure 11. Polar plot of three performance-invariant designs: *A*, *B*, and *C*.

Table III. Comparison of Performance, Cost, and Risk Objective Values for Three Performance-Invariant Designs: A, B, and C[a]

	Performance Objectives		**Cost-Risk Objectives**		
Design	$J_{z,1}$ WFE	$J_{z,2}$ LOS	$J_{cr,1}$	$J_{cr,2}$	$J_{cr,3}$
A: $\mathbf{x}_{iso}^{1}$	20.000	5.2013	$\underline{0.6324}$	0.4668	14.32%
B: $\mathbf{x}_{iso}^{2}$	20.0012	5.0253	0.8960	$\underline{0.0017}$	8.7883%
C: $\mathbf{x}_{iso}^{3}$	20.0001	4.8559	1.5627	1.000	$\underline{5.3067\%}$

[a] Isoperformance tolerance $\tau = 0.05$.

ance levels for these three isoperformance designs are summarized in Table III. Note that these designs are noninferior with respect to each other, similar to the generic isopoints shown in Figure 5(d). A final decision among this small set of isoperformance designs cannot be predicated on further analysis and optimization, but requires stakeholder discussion and consensus. It is in this sense that isoperformance enables engineering insight and acts as an enabling methodology for a target-driven systems engineering process.

5. EXAMPLE 2: TEAM PERFORMANCE IN SPORTS

Systems involving human agents have been traditionally investigated in applied psychology and human factors engineering; see Jones et al. [Kennedy, Turnage, and Jones, 1996; Jones and Kennedy, 2000]. This leads to a probabilistic variant of the isoperformance approach, where contours of equal performance are obtained from empirical models [see Fig. 7(b)]. Models can take on the form of response surfaces such as

$$E[J_{z,i}] = a_0 + a_1(x_{1,i}) + a_2(x_{2,i}) + a_{12}(x_{1,i} - \bar{x}_1)(x_{2,i} - \bar{x}_2) + \cdots, \quad (27)$$

where $E[\;]$ is the expectation operator, $J_{z,i}$ is the performance of the ith individual, team or system, a_0–a_{12} are fitting parameters, $x_{1,i}$ and $x_{2,i}$ are design variables[4] and $\bar{x}_1$ and $\bar{x}_2$ are the mean values of a given data set. Jones and Kennedy [1996] have discussed the problem of finding the isoperformance curves of a baseball team in terms of its final standings (FS = fraction of games won at the end of the regular season) as a function of the team's batting ability (x_1 = RBI = runs batted in) and pitching ability (x_2 = ERA = earned run average).[5] They argue that RBI and ERA can be viewed as independent variables, since the players responsible for achieving these statistics are usually not the same. Teams with a high final standing (> 0.500) are expected to have both good pitching and batting, but for any realistic desired final standing it would be desirable to obtain the trade-off curve between the two factors. The first step is to compile the statistical data and to fit an empirical model to it. The empirical model in the baseball example becomes

$$E[FS_i] = a_0 + a_1(RBI_i) + a_2(ERA_i) + a_{12}(RBI_i - \overline{RBI})(ERA_i - \overline{ERA}). \quad (28)$$

The fitting parameters are obtained by compiling the ERA, RBI, and FS standings from past seasons[6] and optimizing the fitting parameters using least-mean-squares. For the baseball team model [Eq. (28)] we obtain $a_0 = 0.7450$, $a_1 = 0.0321$, $a_2 = -0.0869$, and $a_{12} = -0.0369$. The standard deviation error of the empirical fit is $\sigma_\varepsilon = 0.0493$. The second step is to determine the expected level of performance for team i such that the probability of adequate performance is equal to the specified confidence level. We write

$$E[J_i] = J_{req} + z\sigma_\varepsilon, \quad (29)$$

where $E[J_i]$ is the expected level of performance of team i, J_{req} is the desired (required) final standing at the end

[4] "Determinants" as they are referred to in the Human Factor literature.

[5] The third major category are the fielding statistics, which are ignored here.

[6] The 2000 and 2001 major league baseball (MLB) results are used here (60 data points = 2 seasons × 30 teams).

of the season, z is the confidence level obtained from a normal distribution lookup table of the Gaussian distribution function

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-\frac{z^2}{2}} dz, \tag{30}$$

and σ_ε is the aforementioned model fitting error. This assumes that the error for the empirical model follows a normal distribution. Let the decision-maker (e.g., team owner "i") decide that the required final standing for his team should be $FS_i = 0.550$ and that the probability that this result (performance) should be achieved is 80%. We obtain the expected final standing (target performance) as

$$E[FS_i] = .550 + z\sigma_\varepsilon = .550 + 0.84(0.0493) = 0.5914. \tag{31}$$

In other words, if the final standing FS of team i is to equal or exceed 0.550 with a probability of 80%, then the expected final standing for team i must equal 0.5914. Finally, the isoperformance contour for this performance level (Fig. 12) can be obtained analytically or with one of the algorithms discussed in Section 3.3 as

$$RBI_i = \frac{.5914 - a_0 - a_1 ERA_i + a_{12}\overline{RBI}(ERA_i - \overline{ERA})}{a_1 + a_{12}(ERA_i - \overline{ERA})}. \tag{32}$$

Some interesting conclusions can be drawn from this curve (Fig. 12). The performance measure FS seems to be more sensitive to changes in pitching performance (ERA) than in batting performance (RBI), which supports the commonly held opinion that pitching is more important than batting in major league baseball. Also, as the team goal (FS) becomes more ambitious, the number of options or length of the isoperformance contour becomes smaller. Most to the point, the desired final standing can be achieved with an excellent pitching staff (ERA = 3.0) and modest batting (RBI = 4.2) or conversely with a stellar team at bat and a lesser pitching staff (RBI = 6.0 and ERA = 4.2). It is interesting to note that the best team (Atlanta Braves) and worst team (Chicago Cubs) have nearly identical RBI, but significantly different ERA. The FS for the Braves is somewhat underpredicted by the empirical model due to its inherent stochastic nature. The role of ERA in this situation is reminiscent of the role of I_{sp} in the rocket design example of Section 1.

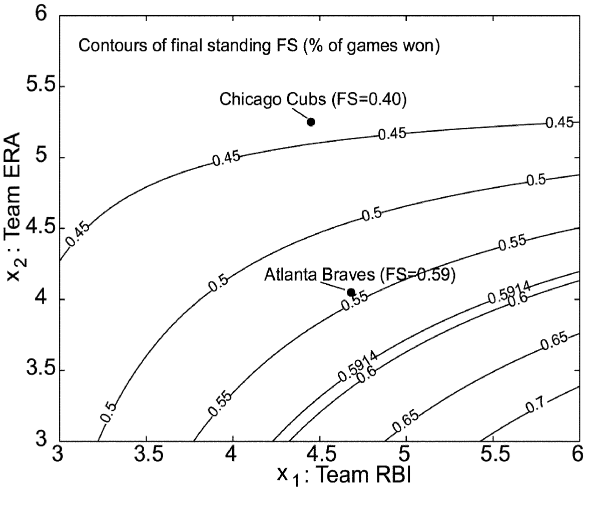


Figure 12. Major league baseball isoperformance analysis (2000–2001 data).

Once an isoperformance curve has been selected, say, 0.5914, as the goal to be achieved, the next step is to decide which points along this curve to pursue. This decision will almost always depend on where the team currently stands with respect to ERA and RBI. If a team already has outstanding pitching but not-so-outstanding batting, it may be easier to find batters who will nudge the team toward its goal than pitchers. It may also be less expensive, since even better pitchers than the team already has are likely to be very expensive. Detailed analyses may reveal many particular ways that a team could produce a better ERA or RBI. Each of these possibilities can be examined with respect to feasibility, cost, and how closely it promises to put the team on the selected isoperformance curve. In the end, of course, management must decide which course to follow, but the isoperformance analysis will have guided the decision-making process from its beginning.

6. SUMMARY AND CONCLUSIONS

Isoperformance is as much a system design philosophy as an operational method. It is true that traditional engineering education and practice heavily emphasize system performance optimization. In reality, however, the notion of optimality for large, complex systems is somewhat less clear. This paper argues that traditional optimization of system performance is not the only reasonable approach in the design and analysis of complex systems. Isoperformance, a complementary approach, does not seek the extremes of system performance, but enforces that the system meet predetermined performance targets (= requirements). This ensures that the system is neither grossly over- nor underdesigned. What can be gained by this approach?

Three potential benefits arise from use of the isoperformance method:

1. It considers not just a single, "optimal" point design but a family of performance-invariant, noninferior designs in terms of other cost and risk criteria.
2. Designs can be found, within the performance invariant set, such that the burden for achieving the system performance is well "balanced" among subsystems.
3. It offers greater insights into the inherent tradeoffs between performance, risk, and cost and allows system analysts and designer to be more interactive, compared to "push-button" optimization.

One may consider performance as a surrogate "currency" for complex systems that are composed of technical and human elements. The fact that suboptimal system performance is acceptable in many cases allows considering the margin between "optimal" performance and the lesser, required performance along the isoperformance contours as a resource. This performance margin can be viewed as a "currency" that can be invested in different ways: making the system more affordable to implement, more robust or flexible, such as enabling upgrades after initial fielding has occurred.

One may argue that similar results could be obtained by an all-in-one multiobjective optimization [Eq. (18)]. The question regarding which of the two approaches, multiobjective optimization or isoperformance, is superior for the design of complex Engineering Systems is not easy to answer or necessarily relevant as they can be viewed as complementary. It is, however, well known that most optimization algorithms experience significant difficulties and computational expense while enforcing equality constraints, and this is the main strength of isoperformance.

The isoperformance approach decouples the problem into three phases that allow system analysts and designers to develop intuition about system tradeoffs that would otherwise remain hidden. The approach does not rely on stakeholder preferences except for the selection of the final design, $\mathbf{x}_{iso}^{**}$, from the efficient set $\mathbf{E}$.

Future work includes extending isoperformance to design problems where both discrete and continuous design variables are present. This is important, as many architectural variables (number of stages in rocket design, Cassegrain versus Gregorian telescope concept, etc.) are discrete in nature and can have a great impact on system performance, cost and risk. Also, of the four tensions shown in Figure 2 we have not explicitly dealt with schedule. To do so would require combining isoperformance analysis and multidisciplinary design optimization with a quantitative model of project management. One could envision a framework where changes in system performance targets are propagated to changes in the associated design variables, which in turn cause schedule impact on the associated design tasks. Work on integrating system design with project management is difficult, but essential for those organizations that strive for excellence and balance across competing demands in complex system design.

NOMENCLATURE

τ	Isoperformance tolerance level (fraction of nominal performance)
d	Exogenous system disturbance input
m	Number of system performance objectives
n	Number of design variables
p	System fixed parameter
r	Reference input signal, Number of cost-risk objectives
u	System control input
w	Filtered disturbance input
$x_i \in \mathbb{R}$	Continuous design variable
y	System output
z	System performance (filtered system output)
G_{sys}, G_I, G_O	System transfer function matrices
J_i, $J_{req,i}$	System objective, required performance level
K	Controller
$\mathbf{p}$	Vector of fixed parameters
$\mathbf{x}$	Design variable vector
$\mathbf{J}$	System objective vector

REFERENCES

H.E. Cook, Product management: Value, quality, cost, price, profits and organization, Chapman & Hall, New York, 1997.

O.L. de Weck and D.W. Miller, Multivariable isoperformance methodology for precision opto-mechanical system, AIAA-2002-1420, 43rd AIAA/ASME /ASCE/AHS Struct Struct Dyn Mater Conf, Denver, Colorado, April 22–25, 2002.

J.J. Gilheany, Optimum selection of dampers for freely vibrating multi-degree of freedom systems, Proc Damping '89, II, 1989, pp. 1–18.

M.B. Jones, Isoperformance and personnel decisions, Hum Factors 42(2) (2000), 299–317.

M.B. Jones and R.S. Kennedy, Isoperformance curves in applied psychology, Hum Factors 38(1) (1996), 167–182.

R.S. Kennedy, J. Turnage,. and M.B. Jones, A meta-model for systems development through life cycle phases—coupling the isoperformance methodology with utility analysis, SAE PAPER 901947, SAE Aerospace Technology Conference and Exposition, Long Beach, CA, 1990.

H.J. Kim, C. Kim, and O.H. Rho, Multipoint inverse design method for transonic wings, J Aircraft 36(6) (1999), 941–947.

I.Y. Kim and O.L. de Weck, Adaptive weighted-sum method for bi-objective optimization: Pareto front generation, Struct Multidisciplinary Optim 29(2) (2005), 149–158.

M.W. Maier and E. Rechtin, The art of systems architecting, 2nd edition, CRC Press, Boca Raton, FL, 2000.

A. Messac, Physical programming: Effective optimization for computational design, AIAA J 34(1) (1996), 149–158.

S.J. Osher and R.P. Fedkiw, Level set methods and dynamic implicit surfaces, Springer, New York, 2002.

V. Pareto, Manuale di economia politica, Societa Editrice Libraria, Milano, Italy, 1906. Translated into English by A.S. Schwier as Manual of political economy, Macmillan, New York, 1971.

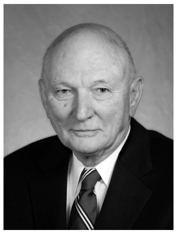
D. Schrage, et al. Current state of the art on multidisciplinary design optimization, An AIAA White Paper, URL reference: http://endo.sandia.gov/AIAA_MDOTC/sponsored/aiaa_paper.html, AIAA Multidisciplinary Design Optimization, Technical Committee, September 1991.

R. Shishko, R.G. Chamberlain, et al., NASA systems engineering handbook, SP-610S, NASA, Washington, DC, June 1995.

H.A. Simon, The sciences of the artificial, 3rd edition, MIT Press, Cambridge, MA, 1996.

K. Tsiolkovsky, The exploration of cosmic space by means of reaction motors, Scientific Review, Moscow and St. Petersburg, 1903.

Olivier L. de Weck is currently an assistant professor with a dual appointment between the Department of Aeronautics and Astronautics and the Engineering Systems Division (ESD) at MIT. His research interests are in Integrated Modeling and Simulation, Multidisciplinary Design Optimization, and System Architecture. In 2001 he obtained a Ph.D. in Aerospace Systems from MIT. From 1987 to 1993 he attended the Swiss Federal Institute of Technology (ETH Zurich) in Switzerland, where he earned a Diplom Ingenieur degree (MS equivalent) in industrial engineering. From 1993 to 1997 he served as liaison engineer and later as engineering program manager for the Swiss F/A-18 program at McDonnell Douglas (now Boeing) in St. Louis, MO. He is serving as the General Chair for the 2nd AIAA Multidisciplinary Design Optimization (MDO) Specialist Conference in 2006 and obtained two best paper awards at the 2004 INCOSE Systems Engineering Conference.



Marshall B. Jones is professor emeritus of Behavioral Science at The Pennsylvania State University and was for 24 years chairman of that department in the College of Medicine. He has conducted extensive research in human factors, psychiatric genetics, especially the genetics of autism, the spread of attitudes and behaviors from one person to another, and the prevention of antisocial behavior. He currently serves as board chairman of Keystone Human Services, a large behavioral health organization headquartered in Central Pennsylvania but active in Connecticut, Maryland, Delaware, and, recently in Eastern Europe, as well as in Pennsylvania. Jones received his B.A. degree from Yale in 1949 and his Ph.D. from the University of California at Los Angeles in 1953.