INCLUDING IMPRECISION IN ENGINEERING DESIGN CALCULATIONS

William S. Law
Erik K. Antonsson
Engineering Design Research Laboratory
Division of Engineering and Applied Science
California Institute of Technology
Pasadena, California

ABSTRACT

The Imprecise Design Tool (IDT) presented in this paper is a working computer implementation of the *method of imprecision*, a formal theory that represents preferences among design alternatives. An aircraft engine design example indicates how the IDT may be applied to support engineering design decisions, using the Engine Development Cost Estimator provided by General Electric Aircraft Engines, Cincinnati, Ohio.

INTRODUCTION

Ullman (1992), among others, notes that computers are not widely used in the early phases of engineering design because (1) existing computer tools need a very refined representation of an object on which to operate, and (2) computers are primarily evaluation tools and of limited value in generating concepts. Preliminary design descriptions are characteristically *imprecise*: the designer has yet to make most of the decisions that will reduce the number of design alternatives considered from many to one.

The Imprecise Design Tool (IDT) is a working computer implementation of the *method of imprecision* (Wood et al., 1992; Otto, 1992), a formal theory for representing and manipulating imprecise preliminary design information. The IDT supports preliminary design decisions which are based on "black box" evaluation tools. A black box could be a computer program, stored data, or a live experiment. Decision-support tools such as the IDT are necessary to effectively evaluate the set of alternative designs for sufficiently complex and imprecise design problems (Figure 1). For extremely complex design problems the computations required to rigorously explore the set of alternative designs can be time-consuming, but advances in the computer field are extending this limit upwards and broadening the range of problems to which decision-support tools can and should be applied.

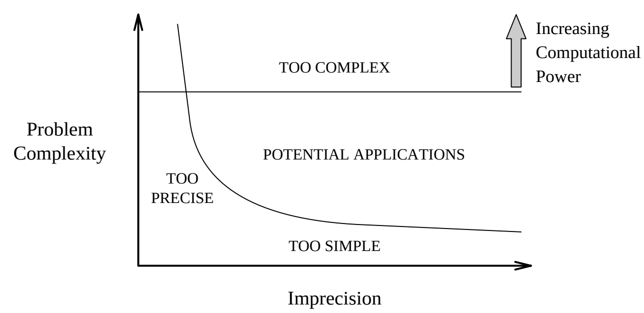

FIGURE 1. POTENTIAL APPLICATIONS FOR THE IMPRECISE DESIGN TOOL.

This paper summarizes the method of imprecision, briefly describes the IDT, and provides an example of how it may be applied to an engineering design problem. See (Law and Antonsson, 1994) for a more detailed description of the IDT. In the example, two imprecisely specified aircraft engine designs will be evaluated and compared using the Engine Development Cost Estimator (EDCE) as the IDT's black box. The example will present some of the difficulties in adapting the method of imprecision to a real engineering problem, as well as practical solutions to these difficulties.

DEFINITIONS AND NOTATION

The *design parameter space* or *DPS* is the set of alternative design configurations considered, described using *design pa-*

rameters which the designer has a direct choice over (Otto, 1992). Design parameters are denoted d_i, where i ranges from 1 to n. The whole set of design parameters is an n vector, $\vec{d}$, and the valid design parameter values within the DPS form a subset $\mathcal{X}$. The set of valid values for d_i is denoted $\mathcal{X}_i$.

The *performance parameter space* or *PPS* is the dependent set of performances evaluated for each design in the DPS, described using *performance parameters* (Otto, 1992). For each performance parameter p_j, where j ranges from 1 to q, there must be a mapping f_j such that $p_j = f_j(\vec{d})$. The set of performance parameters is a q vector, $\vec{p} = \vec{f}(\vec{d})$. The subset of valid performance parameter values $\mathcal{Y}$ is mapped from $\mathcal{X}$ and the set of valid values for p_j is denoted $\mathcal{Y}_j$. For a discussion of using designer parameters and performance parameters to formally model design problems, see Otto (1992).

The subjective satisfaction that a designer has for values of d_i, the ith design parameter, is represented by a membership function on $\mathcal{X}$, termed the *designer preference*:

$$\mu_{d_i}(d_i) : \mathcal{X}_i \to [0,1] \subset \mathrm{R} \tag{1}$$

where the $\mathcal{X}_i$ are assumed to be compact. The customer's satisfaction with values of p_j, the jth performance parameter, is represented by a membership function on $\mathcal{Y}$, termed the *functional requirement*:

$$\mu_{p_j}(p_j) : \mathcal{Y}_j \to [0,1] \subset \mathrm{R}. \tag{2}$$

These preference functions μ_{d_i} and μ_{p_j} are assumed to be monotonically increasing on their support to a range of values (possibly a single value) with peak preference equal to one, and to be monotonically decreasing after the peak.

The combined satisfaction of the designer and customer with a particular design $\vec{d}$ is represented by an overall preference $\mu(\vec{d})$, which is a function of the designer preferences $\mu_{d_i}(d_i)$, and the functional requirements $\mu_{p_j}(p_j)$:

$$\mu(\vec{d}) = \mathcal{P}\left[\mu_{d_1}(d_1), \ldots, \mu_{d_n}(d_n), \mu_{p_1}(f_1(\vec{d})), \ldots, \mu_{p_q}(f_q(\vec{d}))\right]. \tag{3}$$

The *combination function* $\mathcal{P}$ must satisfy continuity and annihilation *i.e.*$(\mathcal{P}[\mu_1, \ldots, 0, \ldots, \mu_{n+q}] = 0)$ (Otto, 1992). Additional restrictions that $\mathcal{P}$ should satisfy for engineering design have been proposed in (Otto and Antonsson, 1991a). The design problem is to maximize μ and so we seek design configurations $\vec{d}^*$ such that:

$$\mu(\vec{d}^*) = \mu^* = \sup\{\mu(\vec{d}) \mid \vec{d} \in \mathcal{X}\}. \tag{4}$$

The peak overall preference in $\mathcal{X}$, μ^*, is equal to the peak overall preference in $\mathcal{Y}$ (Otto et al., 1993).

$\mathcal{P}$ reflects the *design strategy* (Otto and Antonsson, 1991a, 1991b). Suppose that we wish to maximize the lowest preference (μ_{d_i} or μ_{p_j}) for the design. This is a conservative or non-compensating design strategy and $\mathcal{P}$ is min:

$$\mu(\vec{d}) = \min\left[\mu_{d_1}, \ldots, \mu_{d_n}, \mu_{p_1}, \ldots, \mu_{p_q}\right]. \tag{5}$$

Alternatively, we may trade-off different aspects of the design, allowing a more satisfactory aspect to partially compensate for a less satisfactory aspect. This is an aggressive or compensating design strategy and $\mathcal{P}$ is a normalized product:

$$\mu(\vec{d}) = \left(\prod_{i=1}^{n} \mu_{d_i} \prod_{j=1}^{q} \mu_{p_j}\right)^{\frac{1}{n+q}}. \tag{6}$$

These combination functions satisfy continuity and annihilation as well as the restrictions proposed in (Otto and Antonsson, 1991a). Importance weightings may be specified for μ_{d_i} and μ_{p_j} (Otto and Antonsson, 1991b), but they are not relevant to this paper.

THE METHOD OF IMPRECISION

After specifying designer preferences μ_{d_i} on $\mathcal{X}_i$ and functional requirements $\mu_{p_j}(p_j)$ on $\mathcal{Y}_j$, and choosing a design strategy, we begin by determining the induced values of μ_{d_i} on $\mathcal{Y}$, given by the extension principle (Zadeh, 1965):

$$\mu_d(\vec{p}) = \sup_{\vec{d}:\vec{p}=\vec{f}(\vec{d})} \left[\mu_d(\vec{d})\right] \tag{7}$$

where sup over the null set is defined to be zero and $\mu_d(\vec{d})$ is the combined designer preference on $\mathcal{X}$, as distinct from $\mu_d(\vec{p})$, the combined designer preference on $\mathcal{Y}$ which is obtained from $\mu_d(\vec{d})$. $\mu_d(\vec{d})$ arises from splitting the function $\mathcal{P}$ into three operations:

$$\begin{aligned} \mu &= \mathcal{P}_c\left[\mu_d, \mu_f\right] \\ &= \mathcal{P}_c\left[\mathcal{P}_d(\mu_{d_1}, \ldots, \mu_{d_n}), \mathcal{P}_f(\mu_{p_1}, \ldots, \mu_{p_q})\right] \end{aligned} \tag{8}$$

where $\mathcal{P}_d$ combines the designer preferences, $\mathcal{P}_f$ combines the functional requirements, and $\mathcal{P}_c$ combines these subresults. For a conservative design strategy, $\mathcal{P}_d = \mathcal{P}_f = \mathcal{P}_c = \min$. Note that Equation (8) applies on both the DPS and the PPS.

To calculate $\mu_d(\vec{p})$, we use the *Level Interval Algorithm*, or *LIA* (Wood et al., 1992; Otto and Antonsson, 1991b), first proposed by Dong and Wong (1987) as the "Fuzzy Weighted Average" algorithm and also called the "Vertex Method". The LIA uses α–cuts D_{α_k} in $\mathcal{X}$ to calculate induced α–cut intervals in each $\mathcal{Y}_j$, which define the induced α–cuts P_{α_k} in $\mathcal{Y}$:

$$\begin{aligned} D_{\alpha_k} &= \{\vec{d} \in \mathcal{X} \mid \mu_d(\vec{d}) \geq \alpha_k\} \\ &= [d_{1\,\mathrm{min}}^{\alpha_k}, d_{1\,\mathrm{max}}^{\alpha_k}] \times \ldots \times [d_{n\,\mathrm{min}}^{\alpha_k}, d_{n\,\mathrm{max}}^{\alpha_k}] \end{aligned} \tag{9}$$

$$\begin{aligned} P_{\alpha_k} &= \{\vec{p} \in \mathcal{Y} \mid \mu_d(\vec{p}) \geq \alpha_k\} \\ &= [p_{1\,\mathrm{min}}^{\alpha_k}, p_{1\,\mathrm{max}}^{\alpha_k}] \times \ldots \times [p_{q\,\mathrm{min}}^{\alpha_k}, p_{q\,\mathrm{max}}^{\alpha_k}] \end{aligned} \tag{10}$$

where $k = 1, \ldots, M$. After calculating $\mu_d(\vec{p})$, we combine $\mu_d(\vec{p})$ and $\mu_p(\vec{p})$ using Equation 8 to obtain $\mu(\vec{p})$, the overall preference on $\mathcal{Y}$. The set of peak preference performances $\mathcal{Y}^* = \{\vec{p}^* \in \mathcal{Y} \mid \mu(\vec{p}^*) = \mu^*\}$ is found from $\mu(\vec{p})$. The design problem is to find the set of peak preference designs $\mathcal{X}^* = \{\vec{d}^* \in \mathcal{X} \mid \mu(\vec{d}^*) = \mu^*\}$, but without the inverse mapping $\vec{f}^{-1} : \mathcal{Y} \to \mathcal{X}$ we can only obtain $\mu(\vec{d})$ point by point. For a conservative design strategy and a single performance parameter p, we may restrict our search to the α–cut D_{α^*} where α^* is the largest α_k such that $\alpha_k \leq \mu^*$ (Law and Antonsson, 1994).

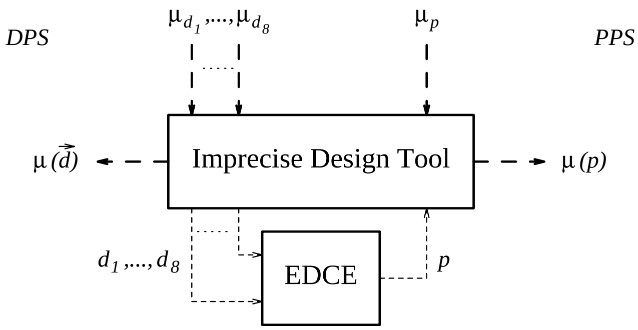

FIGURE 2. THE IMPRECISE DESIGN TOOL AND EDCE.

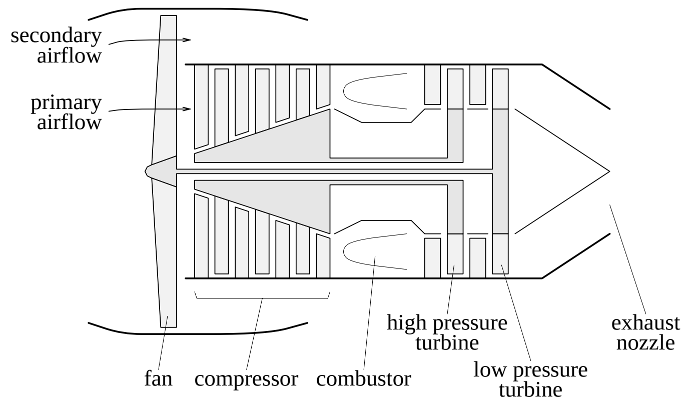

FIGURE 3. SCHEMATIC DIAGRAM OF A TURBOFAN ENGINE.

THE IMPRECISE DESIGN TOOL

The IDT is a C program developed by the authors that implements the method of imprecision with a conservative design strategy ($\mathcal{P} = \min$) for a "black box" mapping $f : \mathcal{X} \rightarrow \mathcal{Y}$. As an example we will use the Engine Development Cost Estimator (EDCE) as the IDT's black box. While EDCE requires crisp inputs and produces a single crisp output, the IDT allows the designer to specify imprecise inputs and obtain an imprecise output (Figure 2). Eight of EDCE's inputs which represent the degree of innovation in eight subsystems of the new engine to be developed were chosen to be design parameters $d_1, \ldots, d_8$. The parameter d_8, for example, corresponds to the fan system. A value of "0%" change indicates that the engine to be developed does not possess a fan system and "10%" change indicates that only support engineering will be required. At the other extreme, "200%" change indicates a new fan with similar or existing technology, fitted to a new engine design. The numeric values of percent change for each of the ten levels defined by EDCE are unimportant: the designer relies on the verbal definition of each level, which is specific to each input. Intermediate values between levels are undefined, and hence the eight inputs are effectively discrete. EDCE produces a single output $p = f(\vec{d})$: an estimate of the development cost for the new engine.

The IDT expects the user to specify designer preferences (μ_{d_i}) at each point in $\mathcal{X}_i$, $i = 1, \ldots, 8$, as an array of numbers. The functional requirement on $\mathcal{Y}$ is specified as an ordered list of pairs (p, μ_p), which define a piecewise linear preference function. The IDT uses a lookup table for $f(\vec{d})$ to avoid repeated EDCE evaluations for the current design calculation and subsequent iterations.

The induced designer preference $\mu_d(p)$ is calculated using the LIA. $\mu_d(p)$ is combined with $\mu_p(p)$ to produce $\mu(p)$, which is saved as an ordered list of pairs (p, μ) defining a piecewise linear preference function on $\mathcal{Y}$. In the same step, the peak preference μ^* and the peak preference set of development costs $\mathcal{Y}^*$ are also found.

After finding μ^*, the IDT determines α^*, the largest $\alpha_k \leq \mu^*$ and $\mu(\vec{d})$ is calculated at every $\vec{d} \in D_{\alpha^*}$ for which there is a lookup table entry for $f(\vec{d})$. Any $\mu(\vec{d}) = \mu^*$ are identified as peak preference design configurations. Where $f(\vec{d})$ is not immediately available, $\mu_d(\vec{d})$ provides an upper bound since, for a conservative strategy, $\mu \leq \mu_d$. We now know $\mu(\vec{d})$ or an upper bound for $\mu(\vec{d})$ at every $\vec{d} \in D_{\alpha^*}$ that could potentially be a peak preference design configuration. The user may also wish to visualize the variation of μ on the DPS, so the IDT allows the selection of points about which it generates eight 2D cross–sections of μ in one design parameter, or four 3D cross–sections of μ in two design parameters.

EXAMPLE

Suppose that exploratory discussions with the customer have culminated in a Request for Proposal (RFP), a requirements document describing the final flying characteristics of an aircraft to be developed. After examination of the RFP and discussions with airframe companies, the design team has decided that a turbofan configuration will be required (Figure 3). There are two options:

1. Developing the new engine from an existing turbojet design by the addition of a front fan with matching shaft and low pressure turbine. The turbojet engine will require minimal redesign to satisfy the RFP, but the addition of a fan, shaft, and low pressure turbine, even if taken from an existing engine, is a major design change.

2. Modifying an existing, but dated, turbofan design. No major design changes will be necessary, but many subsystems will need to be modified.

At this preliminary stage of design, we would like to decide which option to pursue. A key consideration is the total development cost for the engine, which can be estimated by EDCE. The degree of design change that the designers feel will be required in the eight subsystems of the new engine,

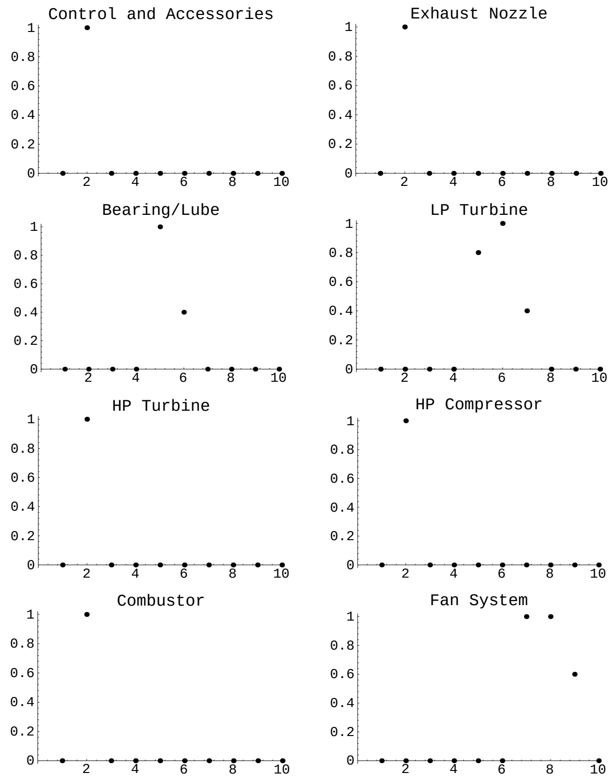

Figure 4. DESIGNER PREFERENCES $\mu_{d_1}, \ldots, \mu_{d_8}$ FOR OPTION 1.

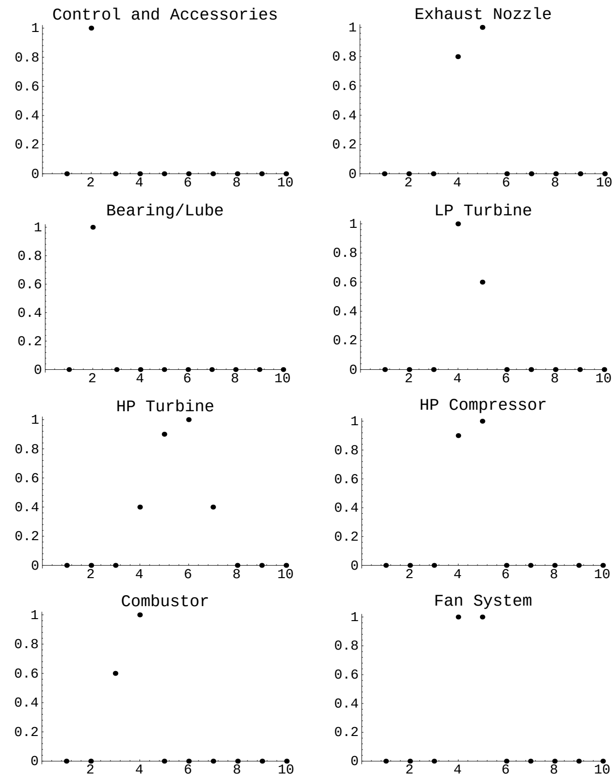

Figure 5. DESIGNER PREFERENCES $\mu_{d_1}, \ldots, \mu_{d_8}$ FOR OPTION 2.

is imprecise, but the IDT allows us to retain the imprecise nature of this design information.

The method of imprecision begins with the specification of designer preferences μ_{d_i} and functional requirement μ_p, and the choice of a design strategy. Here we choose a conservative design strategy. The designer preferences (determined by the design team) for the two options are shown in Figures 4 and 5. The customer wishes to minimize the development cost, suggesting a functional requirement with decreasing preference for increasing cost. For a real design, this decision would depend on many factors, but here we specify a functional requirement that decreases linearly from one, at the minimum development cost (for a turbofan engine with no design modifications), to zero, at the maximum development cost (see Figure 6 or 7). This choice of μ_p, though reasonable, is arbitrary.

$\mu_d(p)$ and $\mu(p)$ calculated by the IDT for each of the two options are shown in Figures 6 and 7. The development costs shown are representative and were not calculated using actual cost data. For option 1, the peak preference μ^* is equal to 0.802 at an estimated development cost of \$185.4 million. For option 2, μ^* is equal to 0.854 at an estimated development cost of \$165.47 million. Option 2 results in a higher peak preference, for the designer preferences, func-

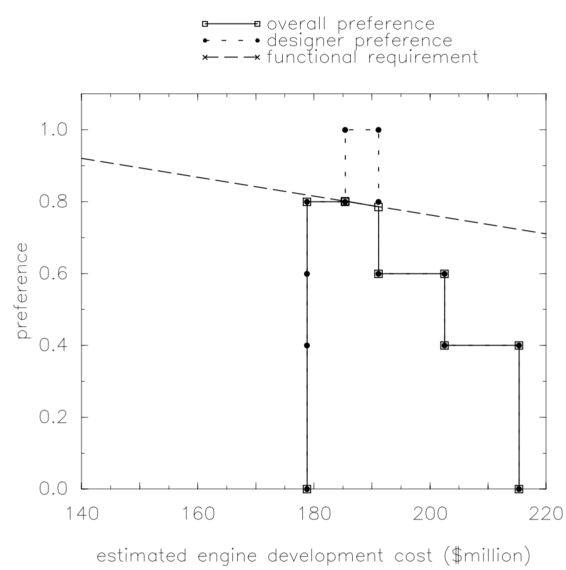

Figure 6. μ, μ_d, AND μ_p ON THE PPS, FOR OPTION 1.

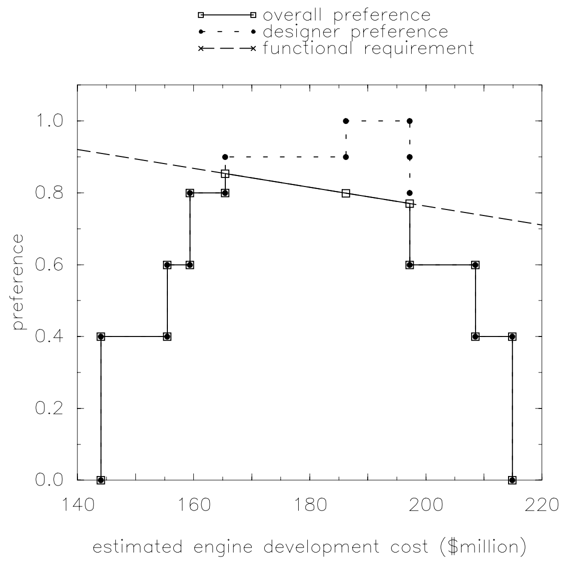

FIGURE 7. μ, μ_d, AND μ_p ON THE PPS, FOR OPTION 2.

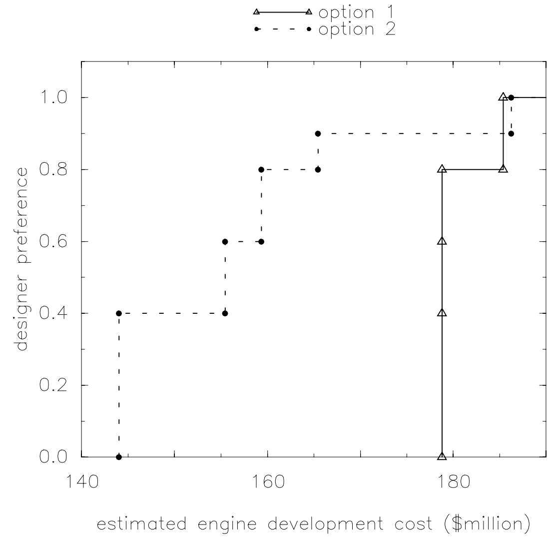

FIGURE 8. μ_d ON THE PPS FOR OPTION 1 AND 2.

tional requirement, and design strategy specified, suggesting that it is the better design choice.

Figures 6 and 7 show that the functional requirement $\mu_p(p)$ attenuates the induced designer preference $\mu_d(p)$ to produce the overall preference $\mu(p)$. This is because we have chosen a conservative strategy for which $\mathcal{P}$ is min. Greater attenuation results in lower peak overall preference μ^* and lower designer preferences for the peak preference designs. As given, $\mu_p(p)$ correctly reflects a relative preference for designs with lower cost p, but $\mu_p(p)$ is an absolute measure. This is an important distinction because the customer's absolute preference is compared to the designer's absolute preference and the min of the two is the measure by which the design is assessed. But we cannot model a customer's absolute preference for cost if it is no more specific than a desire to minimize cost. Let us re-examine our arbitrary choice of the functional requirement μ_p.

Assuming we are only certain of the direction of the slope of $\mu_p(p)$, consider the effect of varying where it intersects the $\mu_d(p)$ pyramid. On the $\mu_d(p) = 1$ plateau $[p^1_{\min}, p^1_{\max}]$, the lowest cost point $p^1_{\min}$ will be preferred when we apply $\mu_p(p)$. All of the pyramid to the right of this point may be ignored, since these points have cost $p > p^1_{\min}$ and preference $\mu(p) \leq 1$. So as we vary $\mu_p(p)$, the peak preference performance traces the left side of the pyramid where $p \leq p^1_{\min}$.

Figure 8 compares $\mu_d(p)$ for the two options. For $\mu_d(p) > 0.9$, option 1 is marginally preferred because it corresponds to a slightly lower development cost: \$185.4 million compared to \$186.25 million. For $\mu_d(p) \leq 0.9$, option 2 is clearly preferred. We may now reconcile the customer's preference for cost with the designer's induced preference for cost in a different way. With a small compromise in designer preference (from $\mu_d = 1$ to $\mu_d = 0.9$), the development cost may be reduced substantially (from \$185.4 million to \$165.47 million). This relaxation in μ_d and in the overall preference μ for the chosen design also corresponds to a change from option 1 to option 2. So assuming that the cost saving justifies the compromise in designer preference we would choose option 2: to develop the new engine from an existing turbojet design, and we would choose the design $\vec{d} = (2, 5, 2, 4, 5, 4, 4, 4)$ with $f(\vec{d}) = \$165.47$ million. This is coincidentally the peak preference design we would have obtained with the $\mu_p(p)$ originally specified.

DISCUSSION

In the design problem presented, EDCE provides only one performance parameter: cost. Since the preference for cost is typically relative (*e.g.*minimize cost), it can be difficult to quantify as an absolute preference curve, as illustrated by our initial choice of the functional requirement μ_p. The example further showed how such a relative preference can be implemented without an absolute preference curve. The IDT can be used to map the designer preferences $\mu_d(\vec{d})$ from the DPS to the PPS, and then the customer's functional requirement may be applied appropriately.

Computational complexity and cost are also significant concerns in using the IDT. For a real design calculation, we would like to minimize the number of EDCE evaluations used to calculate $\mu_d(p)$. As discussed above, we may ignore the points on the right side of the $\mu_d(p)$ pyramid and since we expect to choose designs with high preference, we may also ignore designs with low designer preference $\mu_d(\vec{d})$, thus reducing the number of calculations required.

Gas turbine engine design involves many disciplines. The key stages of the process are thermodynamic cycle analysis, aerodynamic design, and mechanical design (Mattingly et al., 1987). As the design is passed from one group of engineers to another, the mechanical engineers, for example, may discover unacceptably high stresses in the turbine blades and pass the design back to be modified. In practice, imperfect cooperation between the three groups results in many design iterations.

EDCE is part of a larger system of evaluation tools that estimates the total life cycle cost of an aircraft engine. Many of these tools deal with specific subsystems and allow the designer to specify the design in greater detail. Applying the IDT to these evaluation tools would not only allow imprecise designs to be specified, but by extending the notion of combining one set of designer preferences with one set of functional requirements to combining multiple sets of designer preferences and functional requirements, the IDT could support group decisions made collectively by engineers from different disciplines. By requiring the quantification of preferences and the explicit choice of a design strategy, the IDT would expose the decision making process and improve traceability.

CONCLUSION

The method of imprecision, rigorously based on axioms that are consistent with engineering design (Otto, 1992), is a valid methodology for engineering design decisions. The IDT is a working computer implementation of this methodology for a black box evaluation tool such as EDCE. In contrast to other decision-making tools, the IDT allows design imprecision to be explicitly represented and systematically evaluated. The example has shown how the IDT may be used to evaluate and compare two imprecisely described engine designs and how a relative, as opposed to absolute, preference may be represented. Furthermore, by extending the method of imprecision from single to multiple sets of designer preferences and functional requirements, the IDT could be used to support group design decisions.

ACKNOWLEDGMENTS

This material is based upon work supported, in part, by: The National Science Foundation under a Presidential Young Investigator Award, Grant No. DMC-8552695, and NSF Grant No. DDM-9201424. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsors.

The Engine Development Cost Estimator (EDCE) is provided by General Electric Aircraft Engines, Cincinnati, Ohio.

REFERENCES

Dong, W. M., and Wong, F. S., 1987, "Fuzzy Weighted Averages and Implementation of the Extension Principle," *Fuzzy Sets and Systems*, Vol. 21, (No. 2), pp. 183–199.

Law, W. S., and Antonsson, E. K., 1994, "Implementing the Method of Imprecision: An Engineering Design Example," In *Proceedings of the Third IEEE International Conference on Fuzzy Systems* Orlando, Florida. IEEE.

Mattingly, J. D., Heiser, W. H., and Daley, D. H., 1987, *Aircraft Engine Design* AIAA Education Series. AIAA, New York.

Otto, K. N., 1992, *A Formal Representational Theory for Engineering Design* Ph.D. thesis, California Institute of Technology, Pasadena, CA.

Otto, K. N., and Antonsson, E. K., 1991a, "Trade-Off Strategies in Engineering Design," *Research in Engineering Design*, Vol. 3, (No. 2), pp. 87–104.

Otto, K. N., and Antonsson, E. K., 1991b, "Trade-Off Strategies in the Solution of Imprecise Design Problems," In Terano, T., et al. eds., *Fuzzy Engineering toward Human Friendly Systems: Proceedings of the International Fuzzy Engineering Symposium '91, Volume 1*, pp. 422–433 Yokohama Japan. LIFE, IFES.

Otto, K. N., Lewis, A. D., and Antonsson, E. K., 1993, "Determining Optimal Points of Membership with Dependent Variables," *Fuzzy Sets and Systems*, Vol. 60, (No. 1).

Ullman, D. G., 1992, *The Mechanical Design Process* McGraw Hill, New York.

Wood, K. L., Otto, K. N., and Antonsson, E. K., 1992, "Engineering Design Calculations with Fuzzy Parameters," *Fuzzy Sets and Systems*, Vol. 52, (No. 1), pp. 1–20.

Zadeh, L. A., 1965, "Fuzzy Sets," *Information and Control*, Vol. 8, pp. 338–353.