# Computational Methods for Decision Making Based on Imprecise Information

M. Bruns[a], C.J.J. Paredis[a], and S. Ferson[b]

[a]*Systems Realization Laboratory*
*Georgia Institute of Technology*
*mbruns@gatech.edu or chris.paredis@me.gatech.edu*

[b]*Applied Biomathematics*
*scott@ramas.com*

**Abstract:** In this paper, we investigate computational methods for decision making based on imprecise information in the context of engineering design. The goal is to identify the subtleties of engineering design problems that impact the choice of computational solution methods, and to evaluate some existing solution methods to determine their suitability and limitations. Although several approaches for propagating imprecise probabilities have been published in the literature, these methods are insufficient for practical engineering analysis. The dependency bounds convolution approach of Williamson and Downs and the distribution envelope determination approach of Berleant work sufficiently well only for open models (that is, models with known mathematical operations). Both of these approaches rely on interval arithmetic and are therefore limited to problems with few repeated variables. In an attempt to overcome the difficulties faced by these deterministic methods, we propose an alternative approach that utilizes both Monte Carlo simulation and optimization. The Monte Carlo/optimization hybrid approach has its own drawbacks in that it assumes that the uncertain inputs can be parameterized, that it requires the solution of a global optimization problem, and that it assumes independence between the uncertain inputs.

**Keywords:** engineering design, probability box, p-box, uncertainty propagation, imprecision, imprecise probability, Monte Carlo, optimization, interval.

## 1. Introduction

### 1.1. DESIGN DECISION MAKING

Design is the process of converting information about system requirements into a specification of

---

a system that satisfies those requirements. This set of system specifications constitutes a design solution. The space of possible design solutions is unstructured and effectively infinite both in dimension and size. In order to successfully navigate through the structurally complex design space, it is necessary to proceed systematically.

Decision-based design is a useful paradigm for thinking systematically about the design process (Mistree, Smith et al. 1990; Hazelrigg 1998; Marston, Allen et al. 2000). Designers progress through the design process with the help of basically two mechanisms: the generation of design alternatives and decision making. From the decision-based design perspective, the critical elements of the design process are the decisions. Note that decision-based design is not an *approach* to design—it is a *perspective*. That is, from the decision-based design perspective, decisions should be the focus of the designer. Within this perspective, there still exist many different approaches.

Every decision in the design process must be made under some degree of uncertainty. Uncertainty exists when the decision maker (DM) does not know the outcome of at least one decision alternative definitely. The dilemma that uncertainty poses for decision making is clear: different decision alternatives might be preferable in different possible (but uncertain) states of the world.

### 1.2. IMPRECISION IN DESIGN

Since uncertainty strongly influences decision making, and therefore design, it is necessary to study the nature of this uncertainty. Uncertainty is typically divided into two components that we call *variability* and *imprecision*. Although some authors question the philosophical validity of this distinction, it has been argued that such a distinction is useful in practice (Ferson and Ginzburg 1996; Hofer 1996; Winkler 1996; Aughenbaugh and Paredis 2005). Variability corresponds to naturally random behavior of a physical system or process. The standard representation of variability is the probability distribution function.

Many of the uncertainties in engineering design are imprecise. Imprecision is uncertainty due to a lack of knowledge or information (Parry 1996). Imprecision is alternatively referred to as incertitude, but to maintain consistency with past research in the engineering design community we use the term “imprecision” in this paper. The standard representation of pure imprecision is the interval (Kreinovich, Ferson et al. 1999; Muhanna and Mullen 2004). Imprecision arises in design from sequential decision-making, statistical data from finite samples, bounded rationality, and many other sources. For a detailed discussion of the sources of imprecision in engineering design, see the companion paper (Aughenbaugh and Paredis 2006).

Traditional decision analysis assumes precise probabilities. That is, it is assumed that all uncertainty is representable as a precise probability distribution. Because of the high degree of imprecision in engineering design, this assumption is not valid. In order to properly account for

imprecise uncertainties in engineering design, alternative representations, methods for propagation, and decision methods must be developed.

### 1.3. NEED TO PROPAGATE UNCERTAINTY THROUGH PERFORMANCE MODELS

Any method for making decisions under uncertainty must provide three essential tools: (1) a formal representation for uncertain quantities; (2) a method for computing with uncertain quantities; and (3) a decision policy that determines an action under uncertainty. Because of the widespread presence of imprecise uncertainty in engineering design, we seek to develop these three tools for the special case of imprecise probabilities. In particular, we need: (1) a formal representation for imprecise probabilities; (2) a method for computing with imprecise probabilities; and (3) a decision policy that determines the best action given imprecise probabilistic information.

This paper addresses item (2). For insight into the development of representations of imprecise probabilities see (Ferson and Ginzburg 1996; Ferson and Donald 1998; Ferson, Ginzburg et al. 2002). Decision making with imprecise probabilities has been addressed in (Levi 1980; Walley 1991) and with specific emphasis on engineering design in (Aughenbaugh and Paredis 2005; Rekuc, Aughenbaugh et al. 2006).

### 1.4. SUMMARY OF THE LITERATURE

Several solutions to the problem of computing with imprecise probabilities have been proposed in the literature. Although analytical methods exist for a limited class of operations on *precise* random variables (Springer 1979), no work has been done to extend these methods to accommodate *imprecise* random variables. A completely stochastic alternative involves double-loop sampling. The current state-of-the-art methods numerically compute best-possible bounds on the resultant probability distribution of some function of imprecise random variables. While these methods are efficient and accurate, they are not practical for a large class of engineering design problems. The weaknesses of these methods will be discussed in section 2.6.

Computing with imprecise probabilities is a generalization of the problem of computing the convolution of probability density functions where the probability density functions happen to be imprecise. In this paper, we use the term convolution to mean any operation on some set of random variables. Extensive summaries of analytical methods for computing convolutions of random variables is found in the book by Springer (Springer 1979) and in the thesis of Williamson (Williamson 1989).

The most straightforward approach for propagating imprecise probabilities through mathematical models is double-loop Monte Carlo sampling – this is alternatively called two-dimensional, 2-D, or second-order Monte Carlo. A formal description of double-loop sampling is given in section 3.2, and a good review is found in (Hoffman and Hammonds 1994). Modifications to pure double-loop sampling methods are presented in (Hofer, Kloos et al. 2002;

Helton and Davis 2003). Monte Carlo techniques are easy to implement, but for many complex problems, their computational cost becomes prohibitive.

The first efficient numerical approach to the propagation of uncertain quantities was presented by Williamson and Downs in (Williamson 1989; Williamson and Downs 1990). Williamson's work was motivated by the desire to develop numerical methods for precise probabilistic arithmetic, but his methods are compatible with imprecise probabilistic arithmetic. Williamson's methods are referred to as *dependency bounds convolutions* because they result in bounds on the true probability distribution under any possible dependence relation between the uncertain quantities. Dependency bounds are "best-possible" in the sense that the resultant bounds are guaranteed to contain the true resultant distribution, and any reduction of the bounds results in the possible exclusion of the true distribution. The commercially available software Risk Calc 4.0 (Ferson 2002) provides an implementation of the dependency bounds methods.

A very similar approach was developed independently by Berleant in (Berleant 1993; Berleant and Goodman-Strauss 1998). Both Berleant's approach and Williamson's approach discretize probability distribution functions and use maximization and minimization operations to find the best-possible probability bounds on the resultant quantity. Berleant's approach is implemented in the software Statool (Berleant and Cheng 1998; Berleant, Xie et al. 2003). Berleant calls his approach *distribution envelope determination* or DEnv. Regan, Ferson, and Berleant (Regan, Ferson et al. 2004) have shown that DEnv and dependency bounds convolution are equivalent for cumulative distribution functions on the positive reals.

These two approaches are fully sufficient for the propagation of uncertain quantities through functional relationships given explicitly as a sequence of binary operations, but they are insufficient for the computations in most realistic engineering design problems. In section 2, we present a formal statement of an engineering design problem and explain why the available methods are insufficient.

### 1.5. MOTIVATION

The presentation of this paper at this conference is motivated by a desire to facilitate collaboration between the design and reliable engineering computing research communities. While much effort has been expended developing algorithms for computing with uncertain information, much of the results of that effort have been inapplicable to realistic engineering design problems. To resolve this impasse, this paper attempts to present a clear, formal description of a general design problem. Our hope is that those in the reliable engineering community will find further motivation for their research and that we, in the engineering design community, will benefit from their technical expertise.

## 2. Design Computing with Imprecise Uncertainty

In order to understand the computational challenges of using imprecise uncertainties, it is necessary to understand the computations present in the design process. As discussed in section 1, the design process progresses by a sequence of decisions in which the set of design alternatives under consideration is sequentially reduced. We denote a set of design alternatives at step *i* by $\mathbf{D}_i$. In the early stages of design, $\mathbf{D}_i$ is complex and poorly defined. Much of design research focuses on developing heuristics for refining $\mathbf{D}_i$ to a mathematically manageable size and structure. In this paper, we are not concerned with such methods. Instead we assume that $\mathbf{D}_i$ is an interval vector (or hypercube) of dimension *n*, $\mathbf{D}_i \in \mathbb{IR}^n$, where $\mathbb{IR}^n$ is the space of real *n*-dimensional interval vectors: $\mathbb{IR} \equiv \left\{ \left[ \underline{x}, \overline{x} \right] : \underline{x}, \overline{x} \in \mathbb{R}, \underline{x} \le \overline{x} \right\}$. More specifically, we assume that we can write

$$\mathbf{D}_i = \left[ \left[ \underline{d}_1, \overline{d}_1 \right], \left[ \underline{d}_2, \overline{d}_2 \right], ..., \left[ \underline{d}_n, \overline{d}_n \right] \right]$$

where each $\underline{d}_k$ and $\overline{d}_k$ represent the lower and upper bounds of some real, continuous design variable $d_k$. Similarly, for discrete design variables, $\underline{d}_k$ and $\overline{d}_k$ correspond to the smallest and largest of the finite set of alternatives. In this context, the set reduction in each design step, $\mathbf{D}_i \to \mathbf{D}_{i+1}$, corresponds to a decrease in interval width for at least one of the *n* design variables. This process of sequential width reduction converges to a final decision which specifies a precisely defined (singleton) design alternative, $\mathbf{D}^* = \left[ d_1^*, d_2^*, ..., d_n^* \right]$.

Since design computations often involve only a sequence of decisions that are assumed to be decoupled, we focus on the computations involved in a single decision. In the following sections, we will examine in greater detail the mechanics of a single design decision. This involves representing the DM's beliefs and preferences and using performance models to predict how a particular design will satisfy the DM's preferences. This section will close with a precise statement and discussion of the computational problem we hope to solve.

### 2.1. ELEMENTS OF A DESIGN DECISION PROBLEM

A rational decision should reflect the DM's beliefs and preferences. Given a set of beliefs, preferences, and a set of design alternatives, the DM uses some decision policy to determine the preferred decision alternative. It is important here to differentiate a *decision alternative* and a *design alternative*. A decision alternative is any choice that the DM has available at any step in the sequential design process. A design alternative, on the other hand, is any completely specified design. A single decision alternative might correspond to multiple design alternatives. See the companion paper (Aughenbaugh and Paredis 2006) for a more detailed discussion.

A decision policy can be represented by the expression

$$\mathbf{D}_{i+1} = \pi(\mathcal{B}, \mathcal{P}, \mathbf{D}_i)$$

which can be translated into the decision to eliminate the set $\mathbf{D}_e = \mathbf{D}_i \setminus \mathbf{D}_{i+1}$. Here $\mathcal{B}$ is a functional representation of the DM's belief state, $\mathcal{P}$ is a functional representation of the DM's

preference state, and $\pi$ is the decision policy. The set of decision alternatives is the set of all the proper subsets of $\mathbf{D}_i$ excluding the null set.

The belief state, $\mathcal{B}$, is some general multi-valued function that embodies the DM's beliefs about the state of the relevant world at the time of the decision. It is a general but quantifiable measure of the DM's uncertainty about the set of relevant states of affairs. In general, $\mathcal{B}$ is a multi-valued function because of the possibility of imprecision. Realizable relevant states (assumed to be quantifiable) will be represented as scalar vectors, $\mathbf{x} \in \mathbb{R}^n$, where each element, $x_j$, of $\mathbf{x}$ corresponds to some relevant uncertain quantity. The set of all relevant states of affairs will be denoted by $\mathbf{X} = [X_1, X_2, ..., X_n]$. In the context of design, $\mathbf{X}$ can be thought of as the set of variables over which the DM has no control. Mirroring the notation for random variables, uppercase is used to emphasize that the actualized relevant state is an uncertain quantity ranging over the space of possible states of affairs. Note that the DM might choose to model any $X_j$ as certain—that is, $X_j = x_j$ is a known quantity. The most common representation of a belief state is a precise probability measure over the sample space of relevant states of affairs. A precise probability measure is a single-valued function $P: \mathbb{R}^n \to [0,1]$. That is, $P(\mathbf{x}) = p$ such that $p \in [0,1]$.

The preference state, $\mathcal{P}$, is some general multi-valued function that embodies the DM's preferences about possible consequences of the decision. Like $\mathcal{B}$, $\mathcal{P}$ can be multi-valued in order to account for imprecision. The uncertain consequences of a decision are dependent on the actual relevant state of affairs $\mathbf{x}^l = [x_1^l, x_2^l, ..., x_n^l]$ (corresponding to state *l*) as well as the decision, $\mathbf{D}^k = [d_1^k, d_2^k, ..., d_m^k]$ (corresponding to decision *k*), taken. The preference state, $\mathcal{P}$, at the time of the decision is a measure of the value of a particular consequence to the designer. The most common representation of the preference state is a single-valued utility function $U: \mathbb{R}^k \times \mathbb{R}^l \to \mathbb{R}$. That is, $U(\mathbf{D}^k, \mathbf{x}^l) = u$. The utility of a of a particular design, $\mathbf{D}^k$, given some specific outcome, $\mathbf{x}^l$, is deterministic, but since $\mathbf{x}^l$ is uncertain, the utility of $\mathbf{D}^k$ is also uncertain.

Unlike the uncertain state vector, $\mathbf{X}$, the design alternative search space, $\mathbf{D}$, is controlled by the DM. Generally, each $d_i$ in $\mathbf{D}$ might be continuous or discrete and bounded or unbounded. For simplicity, we make the assumption that $\mathbf{D}$ is an interval vector in $\mathbb{IR}^n$ as was discussed at the beginning of this section.

Finally, the decision policy, $\pi$, is a general multi-valued functional mapping from the DM's beliefs and preferences to the set of *non-dominated* decision alternatives $\mathbf{D}_{i+1}$. A non-dominated decision alternative is an alternative that, given some body of information, the DM cannot rationally eliminate. In classical decision theory, $\pi$ is "maximize expected utility." Mathematically, the preferred solution is found as

$$\mathbf{D}^* = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} \left[ \sum_{x_1} ... \sum_{x_m} u(\mathbf{D}^k, \mathbf{x}) \, p(\mathbf{x}) \right]$$

or

$$\mathbf{D}^* = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} \left[ \int_{x_1} ... \int_{x_n} u(\mathbf{D}^k, \mathbf{x}) \, p(\mathbf{x}) dx_n ... dx_1 \right]$$

for discrete and continuous problems, respectively. In this case, $\mathbf{D}_e = \mathbf{D}_i \setminus \mathbf{D}^*$.

Two special cases of the general decision problem should be mentioned. Both of these specific cases make assumptions about the uncertainty of the DM's beliefs and preferences. The decision is deterministic when all beliefs and preferences are certain. In this case, the DM can simply maximize the utility over $\mathbf{D}_i$. The preferred design solution will be $\mathbf{D}^* = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} (U(\mathbf{D}^k))$. The DM selects the design that necessarily results in the best system performance. This case is unrealistic since design decisions always involve uncertainty with regards to beliefs and preferences.

The second special case of a general design decision acknowledges the presence of uncertainty, but represents that uncertainty as precise probability distributions. That is the DM's beliefs are purely probabilistic, and his or her preferences are deterministic. Sampling strategies such as Monte Carlo and Latin Hypercube are well-established and frequently-used solutions for propagating precise probabilistic uncertainty (Fishman 1996). The DM is able to make a decision by maximizing the expected utility of the design through stochastic programming. The resulting design solution will be $\mathbf{D}^* = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} (E[U(\mathbf{D}^k)])$. This case is more realistic than the purely deterministic solution described above but is still an approximation because the DM is not able to account for imprecise information.

Before we can study computational methods for handling imprecise information, we must first make some simplifying assumptions about the representation of uncertain quantities and the decision and performance models to be used.

### 2.2. BELIEFS REPRESENTED AS P-BOXES

An uncertain quantity is some event or variable characterized by sets of possible levels of belief. An uncertain quantity is a more general case of a random variable. Whereas a random variable characterizes a quantity by some precise belief function—namely, a probability distribution function, an uncertain quantity assigns a set of belief functions to a single quantity. For instance, consider a bent quarter. I am uncertain about whether it will land heads-up or tails-up, but until I have seen it flipped many times, I am also uncertain about how probable it is that it will land heads-up or tails-up. I believe that the probability of the bent quarter landing heads-up is less than 0.6 and greater than 0.3. My belief state then corresponds to the interval of probability values between 0.3 and 0.6.

The fundamental types of uncertain quantities are intervals and imprecise probabilities. An interval is a connected set of numbers on the real line. Specifically, the real interval $[a,b] = \{x : x \in \mathbb{R} \text{ and } a \le x \le b\}$. An interval represents a perfectly imprecise uncertain quantity since no assumptions are made about beliefs between the upper and lower bounds. Under interval uncertainty, the DM believes only the fact that the true value of the quantity is constrained by two bounds. When a DM represents an uncertain quantity as an interval he or she has no beliefs about the likelihood of any value in the interval. This is an extreme case—most often the DM does have some beliefs about likelihoods. For extensive discussions of propagating interval uncertainty, see (Moore 1979; Alefeld and Herzberger 1983; Kearfott and Kreinovich 1996).

An imprecise probability is an interval-valued probability measure assigned to an uncertain event—for instance, my beliefs about the bent quarter. Imprecise probabilities are discussed and justified thoroughly in (Walley 1991). The theory of imprecise probabilities developed by Walley extends the operational definition of subjective probabilities to allow for imprecision. The primary advantage of using imprecise probabilities for representing uncertain beliefs is that they allow for the representation of both variability and imprecision.

The probability box (p-box) is a formalism for representing uncertain quantities (Ferson and Ginzburg 1996; Ferson and Donald 1998). The defining characteristic of a p-box are the probability bounds that define upper and lower limits on cumulative probability over the domain of the uncertain quantity. When defining a p-box formally, there are essentially two structures involved: the p-box proper, and the p-box function. The p-box proper, $\boxed{X}$, of some uncertain quantity $X$ defines the p-box as a set of distribution functions constrained by probability bounds and the property of being non-decreasing

$$\boxed{X} = \left\{ F_X(x) : \forall x \in \mathbb{R},\ \underline{F}_X(x) \le F_X(x) \le \overline{F}_X(x) \right\}$$

where $\underline{F}_X, F_X, \overline{F}_X : \mathbb{R} \to [0,1]$, $\underline{F}_X = \underline{P}(X \le x)$ and $\overline{F}_X = \overline{P}(X \le x)$ are the lower and upper cumulative probability bounds, and $F_X$ is non-decreasing with $x$. These probability bound functions are determined by the p-box function. The p-box function is an interval-valued mapping from $x$ to the interval $[0,1]$. We express the p-box function as

$$F_{\boxed{X}}(x) = \left[ \underline{F}_X(x), \overline{F}_X(x) \right]$$

where $\underline{F}_X(x) \le \overline{F}_X(x)$ for all $x$. In other discussions, it might be useful to reverse the order of the bounding distributions in the interval above such that $F_{\boxed{X}}(x) = \left[ \overline{F}_X(x), \underline{F}_X(x) \right]$. In this case, $\overline{F}_X(x)$ denotes the left bound on the p-box and $\underline{F}_X(x)$ denotes the right bound. In other words, upper and lower are defined with respect to $x$ rather than with respect to cumulative probability. For our purposes, however, it is more convenient to interpret upper and lower with respect to probability.

The p-box is general enough to represent intervals, probability distributions, scalars, as well as imprecise probability distributions. An interval $X = [a,b]$ corresponds to the p-box defined by the probability bounds

$$\underline{F}_X(x) = \begin{cases} 0, & x < b \\ 1, & x \geq b \end{cases}$$

and

$$\overline{F}_X(x) = \begin{cases} 0, & x < a \\ 1, & x \geq a \end{cases}.$$

A normally distributed random variable, $X \sim N(\mu,\sigma)$, corresponds to the p-box containing only one cdf, $\boxed{X} = \left\{\Phi_{\mu,\sigma}(x)\right\}$, and the degenerate p-box function with $\underline{F}_X(x) = \overline{F}_X(x) = \Phi_{\mu,\sigma}(x)$ where $\Phi_{\mu,\sigma}(x)$ is the cumulative distribution function of the normal distribution with mean $\mu$ and standard deviation $\sigma$. A scalar, *a*, corresponds to the degenerate p-box function with

$$\underline{F}_X(x) = \overline{F}_X(x) = \begin{cases} 0, & x < a \\ 1, & x \geq a \end{cases}.$$

Finally, and most importantly, the p-box can be used to represent imprecise probability distributions such as $X \sim N\left(\left[\underline{\mu},\overline{\mu}\right],\left[\underline{\sigma},\overline{\sigma}\right]\right)$. Here it is known that the uncertain quantity has normal variability with an imprecise mean, $\mu \in \left[\underline{\mu},\overline{\mu}\right]$, and an imprecise variance, $\sigma \in \left[\underline{\sigma},\overline{\sigma}\right]$. This imprecise probability distribution corresponds to the parameterized p-box $\boxed{X}^P = \left\{F_X(x;\mu,\sigma) = \Phi_{\mu,\sigma}(x) : \mu \in \left[\underline{\mu},\overline{\mu}\right], \sigma \in \left[\underline{\sigma},\overline{\sigma}\right]\right\}$ where the superscript *P* denotes that the p-box is parameterized. It is not meaningful to speak of bounding functions for parameterized p-boxes since the parameterized p-box will not contain all non-decreasing functions between its lower and upper bounding functions. Parameterized p-boxes will be discussed in greater detail in section 3.1.

In this paper, the DM's beliefs are modeled as p-boxes. Relating back to previous notation, the DM's beliefs are represented by $\mathcal{B}(\mathbf{x}) = F_{\mathbf{X}}^{\square}(\mathbf{x})$ where $F_{\mathbf{X}}^{\square}(\mathbf{x})$ represents the joint p-box function for the vector of relevant uncertain quantities. A joint p-box is the imprecise equivalent of a joint distribution function for precise probabilities. There are two steps for justifying this representation. First, the theory of imprecise probabilities is the most fully developed model for imprecise uncertainty. Unlike alternative representations of imprecise uncertainty such as possibilities (Dubois and Prade 1988) or fuzzy sets (Zadeh 1965), imprecise probabilities have a clear operational definition. An operational definition is "a rule which indicates how the mathematical notions are intended to be interpreted (Cooke 2004)." The subjective interpretation of probability provides an operational definition in terms of subjective degree of belief expressed

through a willingness to bet (Savage 1972; de Finetti 1980). Walley extends the subjective interpretation to account for the imprecision between minimum selling prices and maximum buying prices of gambles (Walley 1991). For a criticism of uncertainty models without clear operational definitions, see (Cooke 2004).

The second step in the justification for using p-boxes to represent beliefs is that they are intuitive and used in most of the imprecise probability propagation literature for representing imprecise probabilities. Cumulative probabilities are a straightforward way in which to assign definite probabilities to events. As examples of the common use of p-boxes in the literature, see the work of Williamson and Downs (Williamson and Downs 1990), Ferson (Ferson and Ginzburg 1996; Ferson and Donald 1998; Ferson 2002; Ferson and Hajagos 2004), and Berleant (Berleant 1993; Berleant and Goodman-Strauss 1998; Berleant, Xie et al. 2003; Berleant and Zhang 2004).

### 2.3. UTILITY MODELS REPRESENTED BY BLACK-BOX FUNCTIONS

So far, we have only studied decision policy models in terms of abstract functional mappings from beliefs and preferences to a preferred action. To complete the link from generic decision theory to specific design practice, we must first present and justify several assumptions regarding the mathematical models to be used in design decision making.

For practical reasons, proposed solutions should assume that all mathematical models are black boxes. Although it is not true that engineering models are truly black-boxes, in the sense that nobody knows the mathematical operations inside, it is true that much of engineering practice uses previously developed models as if they were black-boxes. In the future, it is possible that the dependency bounds convolution or the distribution envelope determination methods will be implemented in much of the standard engineering software. At this point in time, however, this is not the case. Although Risk Calc 4.0 (Ferson 2002) and Statool (Berleant, Xie et al. 2003) are useful for propagating imprecise information through algebraic models, much of engineering design practice requires the aid of advanced simulation software such as FLUENT or ANSYS. If the representation of beliefs as p-boxes is to take hold in the engineering design community it is necessary that methods be developed that propagate imprecise beliefs through black-box models developed for advanced software.

### 2.4. DECISION POLICIES FOR IMPRECISE BELIEFS AND PREFERENCES

A rational DM must choose decision alternatives that maximize his or her utility. In the presence of uncertainty, utility is no longer certain. Therefore, in accordance with the axioms of decision theory, the DM should choose the alternative that maximizes his or her expected utility, $E[U]$. If the DM's uncertainty is all due to variability, maximizing expected utility is sufficient. However, in the previous discussion, it has been argued that the DM's beliefs and preferences are imprecise. The presence of imprecision results in intervals of expected utility, $\left[\underline{E}[U], \overline{E}[U]\right]$. While

imprecise beliefs and preferences more accurately reflect the DM's knowledge state, they also complicate considerably the act of decision making. A DM with an imprecise knowledge state needs a more sophisticated decision policy than classical decision theory's prescription of "maximize expected utility." Specifically, imprecise preferences lead to indeterminacy, and indeterminacy results in sets of *non-dominated* decision alternatives. In other words, imprecise preferences result in situations in which rational decision makers cannot choose a single alternative from the set of non-dominated alternatives. Researchers in the imprecise probability community have proposed several decision policies to overcome the indeterminacy in imprecise decision making (Troffaes 2004; Rekuc, Aughenbaugh et al. 2006). Here we limit our discussion to two of these criteria: maximality (Walley 1991) and $\Gamma$-maximin (Berger 1985). Any proposed solution to the problem of computing for design decision making with imprecise uncertainty must be compatible with these decision criteria.

To understand the indeterminacy associated with imprecise knowledge better, consider a simple decision problem in which the DM must select a value for a continuous real-valued design variable, *d*. The DM in this situation can quantify his or her preferences for single values of *d* with an imprecise expected utility function, $E[U(d)] = \left[\underline{E}[U(d)], \overline{E}[U(d)]\right]$. The upper and lower bounds of the DM's utility function are shown in Figure 1.

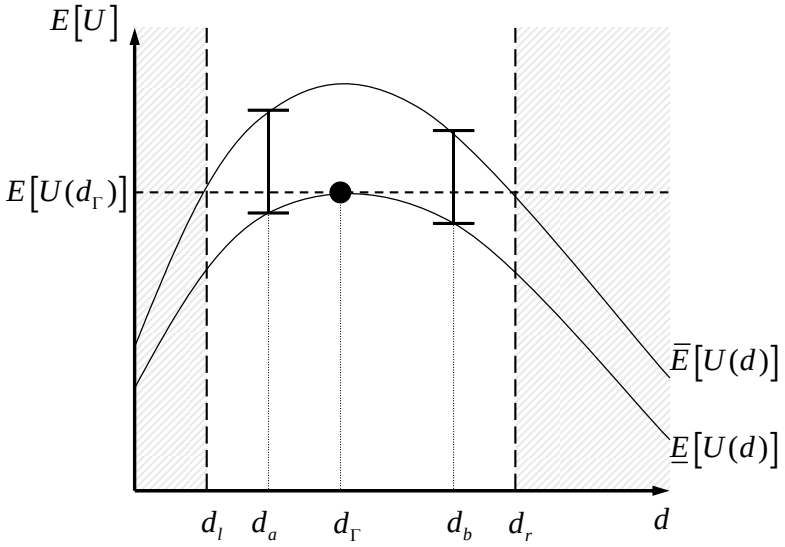


*Figure 1*. Decision indeterminacy with imprecise utilities.

Which value of *d* should the DM select? The higher the utility the more preferred the design, but in this example the utility bounds overlap. Consider a comparison between design alternatives $d_a$ and $d_b$ as shown in Figure 1. The actual utility of either of these alternatives could fall anywhere

between their corresponding upper and lower utility bounds, but the DM has no information about where in those bounds. In some actual cases, $d_a$ will be preferable, but in other cases, $d_b$ will more fully satisfy the DM's preferences. We say that $d_a$ and $d_b$ are *pairwise non-dominated*, and the decision between $d_a$ and $d_b$ is indeterminate. In our example, there is a set of dominated design alternatives. All designs between $d_l$ and $d_r$ are non-dominated by every other design alternative in $[d_l, d_r]$. However, all design alternatives outside of this region are pairwise dominated by the design alternative $d_\Gamma$. Therefore, a rational DM will eliminate the set of design alternatives $d < d_l$ and $d > d_r$. However, indeterminacy remains for all designs between these two bounds. In engineering design, indeterminacy is ultimately not an option since a final design for production cannot be imprecisely specified. Therefore, the DM needs a more sophisticated decision policy in order to further distinguish the space of decision alternatives.

Indeed there is no decision policy that is able to identify a *single* rationally preferred solution in the presence of imprecise uncertainty because indeterminacy is inherent in the problem. Maximality can be used to reduce the size of the set of rational decision alternatives—the DM could rationally choose *any* of the alternatives in that set, but none of the decision alternatives in that reduced set is rationally preferable to any of the others in that set given the current knowledge state of the DM. Decision policies for imprecise uncertainty can be grouped into two general strategies: (1) those that seek to minimize the size of the set of non-dominated alternatives through more sophisticated comparisons of alternatives, and (2) those that select a single-valued solution based on some semi-arbitrary decision criterion. While strategies of type (1) are preferable for rational decision making, for practical purposes, the DM may need to employ some strategy of type (2) in order to find a single-valued design solution.

The decision policies that seek to minimize the set of non-dominated alternatives differ in the amount of information they take into account. Generally, as more information is considered, the resultant set of non-dominated alternatives will decrease in size. The *maximality* criterion (Walley 1991) is well-suited for a broad-class of decision problems because it takes into account most of the available relevant information. By introducing differences in expected utility, the DM is able to identify alternatives that are dominated throughout the entire space of possible states of affairs, **X**. A strict comparison of utility bounds will lose this additional information. The maximality criterion takes into account shared uncertainty. Shared uncertain variables, $z_s$, are those uncertain quantities that are independent from the design variables—i.e. no matter what design variable is selected the shared uncertain variable will take the same unknown value. Therefore, when comparing the utility of two designs, the DM should evaluate both utilities at the same values of the shared uncertain variables. The maximality criterion prescribes that the DM eliminate all decision alternatives for which, when compared to some other alternative evaluated at the same values for the shared uncertain variables, the upper bound on their expected difference in utility is strictly less than zero. Formally,

$$\mathbf{D}_e = \left\{ \mathbf{D}^j \in \mathbf{D}_i : \left(\exists \mathbf{D}^k \in \mathbf{D}_i\right) \left( \max_{\substack{z_s \in Z_s \\ z_j \in Z_j \\ z_k \in Z_k}} \bar{E}\left[ U\left(\mathbf{D}^j, z_j, z_s\right) - U\left(\mathbf{D}^k, z_k, z_s\right) \right] < 0 \right) \right\}$$

where $\mathbf{D}^j$ and $\mathbf{D}^k$ are specific decision alternatives in the set $\mathbf{D}_i$, $Z_s$ is the set of shared uncertain variables, $Z_j$ is the set of uncertain variables specific to $\mathbf{D}^j$, and $Z_k$ is the set of uncertain variables specific to $\mathbf{D}^k$. As an illustration of the use of the maximality criterion, consider again the example in which the DM is trying to select a single value for *d*. Based on past experience, or some other heuristic, the DM believes that $d^*$ will most likely be the preferred solution. In order to eliminate a larger set of design alternatives, the maximality criterion requires that the DM calculate $\bar{E}\left[U(d_i, Z_s, Z_i) - U(d^*, Z_s, Z^*)\right]$ for all $d_i \neq d^*$. A plot of this expected difference in utility is shown in Figure 2.

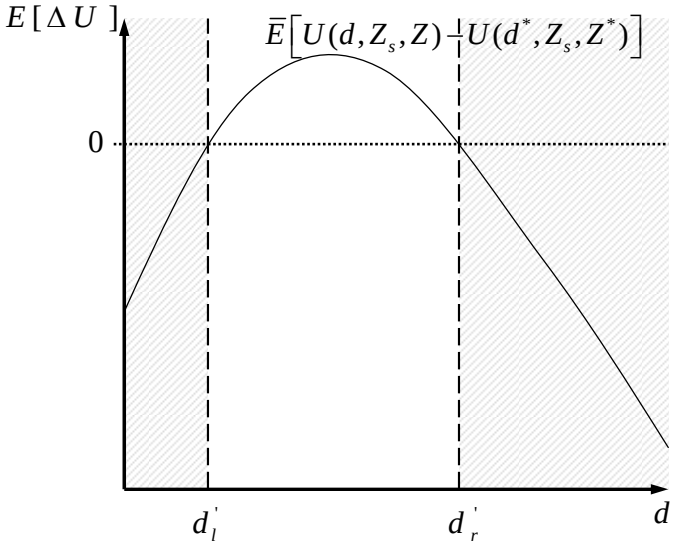


*Figure 2.* Elimination with the maximality criterion.

For all values of *d* less than $d_l'$ and greater that $d_r'$, $\bar{E}\left[U(d_i, Z_s, Z_i) - U(d^*, Z_s, Z^*)\right] < 0$. This means that no matter what the actual relevant state of affairs, $d^*$ will outperform those designs, and these regions can be eliminated from consideration. In terms of previous notation, $\mathbf{D}_e = \left\{ d : d < d_l' \text{ and } d > d_r' \right\}$. While application of the maximality criterion will identify a smaller

set of non-dominated alternatives, the DM will still remain indeterminate between the reduced set of alternatives—in this example the DM is indeterminate between all $d \in \left[d_l', d_r'\right]$. In general, the bounds found through application of the maximality criterion will be tighter than the bounds arising from the application of the interval dominance criterion—that is, $\left[d_l', d_r'\right] \subseteq \left[d_l, d_r\right]$ and most often $\left[d_l', d_r'\right] \subset \left[d_l, d_r\right]$.

The use of shared uncertain variables is similar to the variance reduction technique of using common random numbers (CRNs) in simulation (Law and Kelton 2000). The goal of a simulation is usually to compare two scenarios or alternative designs by examining the difference in output for different combinations of control parameters. If different random numbers are used in the simulations for the different alternatives, additional noise is introduced into the model. CRNs are used to induce correlation between scenarios, thereby reducing the variances of the results. In engineering design, shared uncertainty is an inherent characteristic of the problem. Therefore, a DM does not have to add the commonality, he or she merely needs to recognize it and take advantage of that additional property when it exists. The maximality criterion is a means of exploiting this inherent commonality. A detailed discussion of shared uncertainty can be found in the Master's Thesis of Rekuc (Rekuc 2005) as well as in (Rekuc, Aughenbaugh et al. 2006).

In order to identify a single-valued decision, the DM must employ some semi-arbitrary decision policy. The most conservative of these types of policies is the $\Gamma$-maximin criterion (Berger 1985). Very simply, $\Gamma$-maximin prescribes that the DM select the alternative that maximizes the lower bound on expected utility. In other words, the DM selects the best worse case solution. Formally, the $\Gamma$-maximin solution is found by the expression

$$\mathbf{D}^{\Gamma} = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} \left( \underline{E}_{\boxed{\mathbf{X}}} \left[ U\left(\mathbf{D}^k, \mathbf{x}\right) \right] \right)$$

where the subscript, $\boxed{\mathbf{X}}$, on $\underline{E}$ denotes that the lower expectation is taken over the entire uncertain state space. In Figure 1, the $\Gamma$-maximin solution is marked $d_{\Gamma}$. Selecting the $\Gamma$-maximin solution assures that in the worst-case actualized state of affairs, $d_{\Gamma}$ will outperform any other design alternative operating in its worst-case actualized state of affairs. This is semi-arbitrary because the DM has no rational reason to believe that the worst-case will be actualized, but the DM can still be certain that performance will at least exceed $\underline{E}\left[U(d_{\Gamma})\right]$.

In the presence of imprecision, the DM will generally need to resort to using some semi-arbitrary decision policy such as $\Gamma$-maximin to make a final decision. What value then are the interval dominance and maximality criteria? Should not the DM just compute and select the $\Gamma$-maximin solution? The $\Gamma$-maximin solution is a function of the body of information available to the DM. Since the design process is not self-contained, this body of information is not static. As the DM progresses through the design process, new information about the structure of the design space and the likelihood of different relevant states of affairs become known. Therefore, the DM should delay making unnecessary (i.e. specific) decisions in the early stages of the design

process. The value of proceeding through the design process with sets of design alternatives is discussed in the set-based design literature (Sobek and Ward 1996; Sobek, Ward et al. 1999; Rekuc, Aughenbaugh et al. 2006). The maximality criterion leads to tight, but rational bounds, on the most-preferred solution and so it is therefore useful in the early stages of the design process.

## 2.5. PROBLEM STATEMENT

Now that the general issues involved in computing with imprecise information have been explicated, we can now present a concise statement of the problem.

*Given:*

1. A utility black-box function $U = f(\mathbf{D}^k, \mathbf{x})$ where $U$ is the utility of the design $\mathbf{D}^k \in \mathbb{R}^m$ dependent on some $\mathbf{x} \in \mathbb{R}^n$. Generally, $f$ is an interval-valued mapping $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{IR}$ resulting in the lower and upper utilities $\underline{U}(\mathbf{D}^k, \mathbf{x})$ and $\overline{U}(\mathbf{D}^k, \mathbf{x})$.

2. A vector of p-boxes of dimension *n*, $\boxed{\mathbf{X}} = \left[ \boxed{X}_1, \boxed{X}_2, ..., \boxed{X}_n \right]$, describing the uncertainty about the relevant state of affairs, **x**. This assumes that no *joint* p-box distribution is known which is typically the case in engineering problems. In other words, nothing is known about the dependence relationships between the uncertain quantities.

*Find:*

1. The lower and upper expected utilities of a design, $\mathbf{D}^k$, with respect to the vector of uncertain quantities, $\boxed{\mathbf{X}}$: $\underline{E}_{\boxed{\mathbf{X}}}\left[U\left(\mathbf{D}^k, \mathbf{x}\right)\right]$ and $\overline{E}_{\boxed{\mathbf{X}}}\left[U\left(\mathbf{D}^k, \mathbf{x}\right)\right]$.
2. The set of dominated solutions under the maximality criterion: $$\mathbf{D}_e = \left\{ \mathbf{D}^j \in \mathbf{D}_i : \left(\exists \mathbf{D}^k \in \mathbf{D}_i\right) \left( \max_{\substack{z_s \in Z_s \\ z_j \in Z_j \\ z_k \in Z_k}} \overline{E}\left[ U\left(\mathbf{D}^j, z_j, z_s\right) - U\left(\mathbf{D}^k, z_k, z_s\right) \right] < 0 \right) \right\}.$$
3. The $\Gamma$-maximin solution: $\mathbf{D}^{\Gamma} = \arg\max_{\mathbf{D}^k \in \mathbf{D}_i} \left( \underline{E}_{\boxed{\mathbf{X}}}\left[U\left(\mathbf{D}^k, \mathbf{x}\right)\right] \right)$.

## 2.6. INADEQUACIES OF THE AVAILABLE METHODS

In section 1.4, four approaches for propagating uncertainty were discussed—1) distribution convolutions, 2) dependency bound convolutions, 3) distribution envelope, and 4) double-loop sampling. At present, the first three of these approaches are incapable of solving the problem stated in section 2.5. The fourth approach works but is relatively inefficient. Before an alternative method is proposed, it is necessary to explain why the existing methods fail.

The ideal solution would be to formulate and analytically solve the appropriate set of distribution convolutions. Unfortunately, this is practically impossible. The transformation methods described in Springer (Springer 1979) are limited to basic binary algebraic operations for independent variables with a few distribution shapes. Yager's method (Yager 1986) also requires independent distributions, although it can handle arbitrary shapes and operations. All these methods are impossible or very cumbersome for black-box computer models where the functional relationship is not given explicitly as a sequence of binary operations. Analytical methods appear even less tractable in the presence of imprecision where *sets* of distributions must be convolved.

The dependency bounds approach of Williamson and the distribution envelope approach of Berleant are considerably more promising, but they must overcome at least two obstacles before they can be used in engineering design. First, both of these approaches depend strongly on the methods of interval arithmetic for which the presence of repeated variables can result in over-conservative (i.e. not best-possible) solution bounds. While sub-interval reconstitution methods work well for low-dimensional problems (Moore 1979; Ferson and Hajagos 2004; Ferson, Nelsen et al. 2004), they are prohibitively expensive in realistic engineering problems with a large number of imprecise quantities. Second, black-box propagation of intervals is still only workable for quasi-linear problems. Trejo and Kreinovich have developed a randomized algorithm for propagating interval uncertainty through black-box models (Trejo and Kreinovich 2001; Kreinovich and Ferson 2004), but the method assumes that the black-box model is broadly linear in the region of sampling. It is unclear at this point if this black-box method has general applicability towards complex engineering analysis models.

In order for the dependency bounds and distribution envelope methods to be applicable for engineering design, methods for better propagating intervals through black-box models in the presence or many repeated variables need to be developed. If these conditions were met, it would then be necessary to convince the producers of the standard engineering analysis software to incorporate these methods into their products. While this seems possible, and is perhaps the most desirable solution, our concern is more immediate: how can engineers use the tools available to them today to make realistic design decisions under imprecise uncertainty?

One very simple and easy-to-implement approach is a double-loop sampling routine. A formal discussion will be presented in section 3, but double-loop sampling involves random sampling across the two dimensions of an uncertain quantity (Hoffman and Hammonds 1994; Helton and Davis 2003). Since sampling routines only require evaluations at scalar values of the

set of uncertain variables, these approaches meet our requirement of being compatible with black-box utility models. For high-dimensional problems, double-loop sampling can become prohibitively expensive because the sampling in the outer loop does not retain the computational advantages of Monte Carlo simulation. Specifically, the outer loop sampling is not used to determine an expected value but rather the extrema of results of the inner loop. To approximate these bounds accurately an increasingly large number of samples must be taken as the dimensionality of the problem increases. As a possible solution to this, some authors have suggested a sensitivity analysis approach (Hofer, Kloos et al. 2002). In the next section, we present an alternative means of speeding up double-loop sampling in which one of the sampling loops is replaced by an optimization algorithm.

## 3. Optimizing over Imprecise Distribution Parameters

Of the available methods, double-loop sampling is the only solution convenient for functioning through a black-box utility model. For problems of high-dimensionality, though, double-loop sampling can be prohibitively expensive. In this section, a modification of double-loop sampling is proposed in order to attain a more efficient method for computing with uncertain quantities through black-box utility models.

### 3.1. PARAMETERIZED P-BOXES

In order to clarify the discussion, it is first necessary to present a simplified representation of the general p-box presented in section 2.2. A parameterized p-box is the set of all possible distributions resulting from some known distribution function with imprecisely known parameters. Formally,

$$\boxed{X}^P = \left\{ F_X(x;\boldsymbol{\theta}) : \boldsymbol{\theta} \in \left[ \underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}} \right] \right\}$$

where $F_X(x;\boldsymbol{\theta})$ is non-decreasing with x, and $\boldsymbol{\theta} \in \mathbb{R}^q$ is a vector of distribution parameters that affect the shape or scale of $F_X$. Imprecision is introduced through uncertainty in the parameters. Specifically, the DM is uncertain of the true values of the distribution parameters except for the fact that they lie within known bounds. That is, for all $\theta_k \in \boldsymbol{\theta}$, $\underline{\theta}_k \le \theta_k \le \overline{\theta}_k$.

It is important to emphasize the difference between a parameterized p-box and a general p-box. Similar to a general p-box, a parameterized p-box is a set of non-decreasing probability distribution functions constrained by upper and lower bounds. But unlike a general p-box, a parameterized p-box does not contain all possible non-decreasing distributions lying between its lower and upper bounds. In set notation, if $\boxed{X}$ and $\boxed{X}^P$ share the same bounding functions,

then $\boxed{X}^P \subset \boxed{X}$. To see this, consider a p-box and a parameterized p-box with the same upper and lower bounds.

$$\boxed{X} = \left\{ F_X(x) : \underline{F}_X(x) \le F_X(x) \le \overline{F}_X(x) \right\}$$

where $\underline{F}_X$ is normally distributed with mean $\mu = 4$ and standard deviation $\sigma = 1$ and $\overline{F}_X$ is normally distributed with $\mu = 1$ and $\sigma = 1$. A parameterized p-box with identical bounds is

$$\boxed{X}^P = \left\{ F_X(x;\mu,\sigma) : X \sim Normal\left(\mu = [1,4], \sigma = 1\right) \right\}.$$

Both of these sets of functions are constrained by the bounds $\underline{F}_X$ and $\overline{F}_X$, but $\boxed{X}$ contains functions not found in $\boxed{X}^P$ as shown in Figure 3.

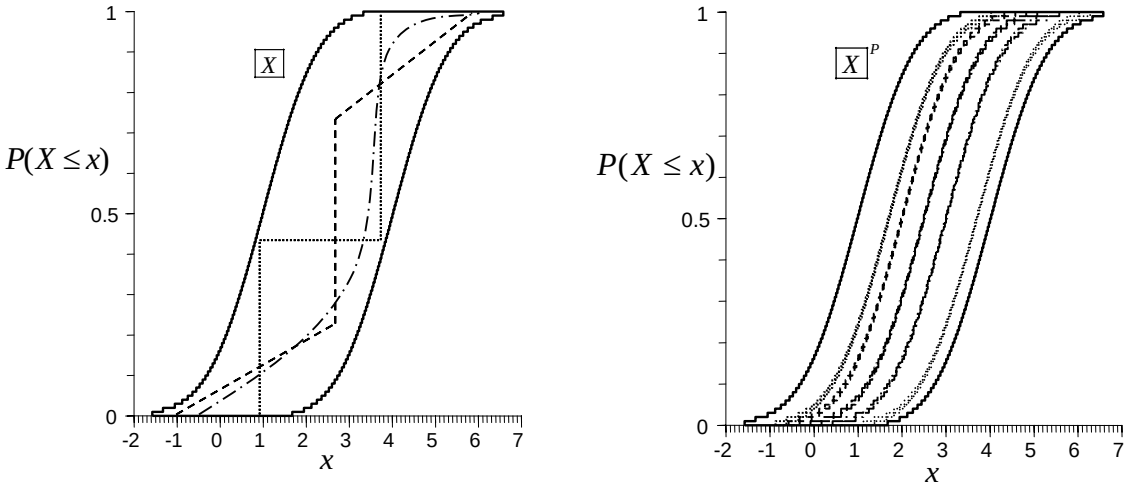


*Figure 3*. Comparison of general and parameterized p-boxes.

Though less general, a parameterized p-box is in many cases a better representation of the DM's beliefs about an uncertain quantity. A common example of this arises in statistical parameter estimation where data gives rise to confidence intervals on the true parameter values for some random variable with a known distribution. If the DM's beliefs cannot be represented as parameterized p-boxes, then double-loop sampling cannot propagate those beliefs.

### 3.2. DOUBLE-LOOP SAMPLING

Double-loop sampling involves two layers of sampling: one associated with distribution parameters and the other associated with the distributions themselves. In effect, double-loop sampling involves sampling from sampled distributions.

Recall that our problem is to determine $\underline{E}_{\boxed{\mathbf{X}}}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right]$ and $\overline{E}_{\boxed{\mathbf{X}}}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right]$ for a given design, $\mathbf{D}^k$. The DM has some black-box utility function, $\left[\underline{U},\overline{U}\right]=f\left(\mathbf{D}^k,\mathbf{x}\right)$, that computes an interval of utility for a given design, $\mathbf{D}^k$, and a precisely known state of relevant affairs, $\mathbf{x}$. Assuming that the DM's belief state is representable by a vector of parameterized p-boxes, $\boxed{\mathbf{X}}^P$, the DM can determine the upper and lower expected utilities of design $\mathbf{D}^k$ with double-loop sampling. The outer loop will be called the *"parameter" loop* since it involves sampling different values for the set of distribution parameters for all of the uncertain quantities. The inner loop will be called the *"probability" loop* since it involves sampling from precise probability distribution functions.

The first step in double-loop sampling is to define a vector containing all distribution parameters for all of the uncertain quantities. Each $\boxed{X}_j^P \in \boxed{\mathbf{X}}^P$ has associated with it a set of imprecise parameters stored in the vector $\boldsymbol{\theta}^j \in \left[\underline{\boldsymbol{\theta}}^j,\overline{\boldsymbol{\theta}}^j\right]$. The number of parameters associated with a single uncertain quantity, $x_j$, is denoted $q_j = length\left(\boldsymbol{\theta}_j\right)$. For notational convenience, it is desirable to combine each of these $q_j \times 1$ vectors into a single vector representing all relevant distribution parameters. This super-vector will be denoted $\boldsymbol{\Theta}$. Also, by extension from the lower and upper bounds of the sub-vectors, lower and upper bounds of the super-vector can be determined. That is, the vector of distribution parameters is constrained such that $\boldsymbol{\Theta} \in \left[\underline{\boldsymbol{\Theta}},\overline{\boldsymbol{\Theta}}\right]$. These parameter bounds are important as they represent all of the imprecision in the DM's belief state. The purpose of the parameter loop is to experiment with these imprecise distribution parameters in order to approximate the smallest and largest utility that the DM should expect for design $\mathbf{D}^k$.

In the parameter loop, the space of the parameter vector, $\boldsymbol{\Theta}$, is explored by random sampling. Once the DM has defined the elements in $\boldsymbol{\Theta}$, he or she must first randomly select a single point in the space of $\boldsymbol{\Theta}$. This point corresponds to a set of precise distributions for all uncertain quantities and will be denoted $\boldsymbol{\Theta}^a$.

The probability loop uses these precise distribution functions to solve a purely probabilistic sampling problem. Specifically, the probability loop uses Monte Carlo samples from the distributions defined by $\boldsymbol{\Theta}^a$ to compute an expected utility of the design $\mathbf{D}^k$. The expected utility

of $\mathbf{D}^k$ given some $\boldsymbol{\Theta}^a$ is denoted $E_a$. The process of computing an $E_a$ is repeated for $s$ randomly sampled points, $\boldsymbol{\Theta}^a$, in the parameter space, $\boldsymbol{\Theta}$. That is, the DM computes an $E_a$ corresponding to some $\boldsymbol{\Theta}^a$ for $a = 1,...,s$.

If the sampled parameter vectors sufficiently cover the parameter space, then the largest and smallest values of the set $\{E_a\}$ can be used to approximate the lower and upper expected utilities of the design $\mathbf{D}^k$. Formally, the lower and upper expected utilities are approximated by the expressions $\underline{E}_{[\mathbf{X}]^p}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right] \approx \min_{1\le a\le s} E_a$ and $\overline{E}_{[\mathbf{X}]^p}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right] \approx \max_{1\le a\le s} E_a$. A schematic of the double-loop sampling process is sketched in Figure 4.

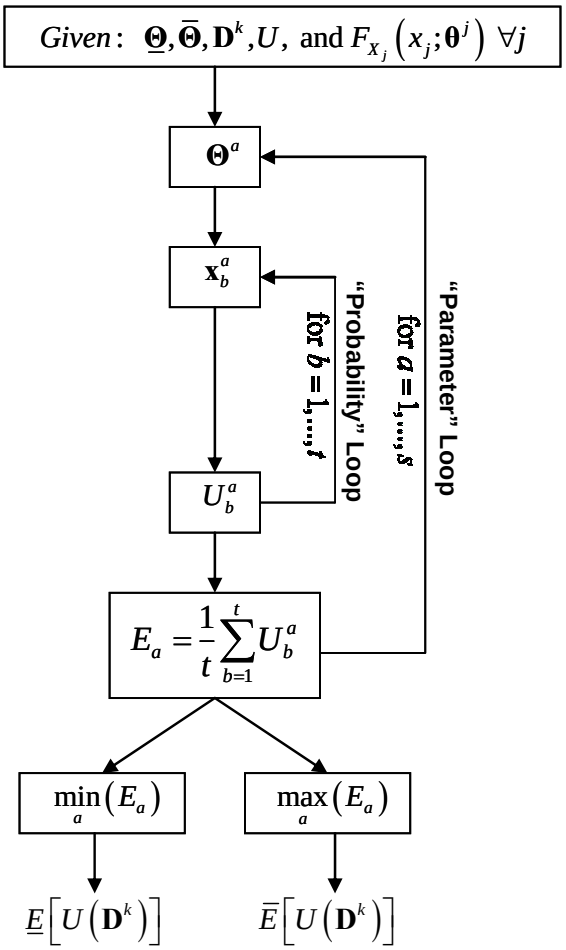


*Figure 4.* Diagram of double-loop sampling method.

### 3.3. OPTIMIZING IN THE "PARAMETER" LOOP

In an attempt to overcome the intractability of double-loop sampling for high-dimensional problems, we propose replacing the sampling in the "parameter" loop with an optimization algorithm. The "parameter" loop optimizer is used to locate the points $\boldsymbol{\Theta}^l$ and $\boldsymbol{\Theta}^u$ in the parameter space that result in the smallest and largest expected utilities, $E_l$ and $E_u$. These utilities are then used to approximate the lower and upper expected utilities for design $\mathbf{D}^k$:

$$\underline{E}_{[\mathbf{X}]^P}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right] \approx E_l$$

$$\overline{E}_{[\mathbf{X}]^P}\left[U\left(\mathbf{D}^k,\mathbf{x}\right)\right] \approx E_u .$$

Essentially, the modified double-loop sampling method is the same as in pure double-loop sampling except that $\boldsymbol{\Theta}^a$ is updated intelligently. The modified approach requires the solution of the following two optimization problems:

$$(1)\ \underset{\Theta\in[\underline{\Theta},\overline{\Theta}]}{\text{minimize}}\ E = f\left(\boldsymbol{\Theta}\right) \Rightarrow E_l$$

$$(2)\ \underset{\Theta\in[\underline{\Theta},\overline{\Theta}]}{\text{maximize}}\ E = f\left(\boldsymbol{\Theta}\right) \Rightarrow E_u$$

where $\underline{\boldsymbol{\Theta}}$ and $\overline{\boldsymbol{\Theta}}$ are the upper and lower bounds on the parameter space.

Numerically, solving these optimization problems poses two challenges: the objective function, $E = f\left(\boldsymbol{\Theta}\right)$, 1) is approximated non-deterministically and 2) could have local extrema. Different random deviates in the "probability" loop will result in different approximations of $E_a$ for a given vector of parameters, $\boldsymbol{\Theta}^a$. For gradient-based optimizers, this is problematic since the approximation to the objective function will develop sharp local gradients. One possible solution to challenge 1) is to use the same set or random deviates for each step of the optimization algorithm. This of course introduces a bias into the resulting $E_l$ and $E_u$, but this bias can be made arbitrarily small by increasing the number of "probability" loop samples, *t*. The second challenge to solving these optimization problems is due to the nature of the true objective function. For realistic engineering problems, $E = f\left(\boldsymbol{\Theta}\right)$ is often multi-modal. One possible solution to challenge 2) is to repeat the optimizations from multiple starting points, $\boldsymbol{\Theta}^1$. Both of these solutions have proven to be effective in the design of an off-road vehicle gearbox (Rekuc, Aughenbaugh et al. 2006).

### 3.4. WEAKNESSES OF THE ALGORITHM

For many problems, the modified double-loop sampling algorithm described above will more efficiently locate the minimal and maximal sets of distribution parameters. However, the

modified approach retains some of the weaknesses of the pure double-loop sampling and even introduces some new difficulties.

Like pure double-loop sampling, the modified approach assumes a known dependence between the uncertain quantities involved in the computation. If a parameterized joint distribution function of all uncertain quantities were available, it would be compatible with either approach, but for practical problems it is almost never the case that the DM knows a fully characterized joint distribution. The dependency bounds approach, as described in section 1, makes no assumptions about the dependence between uncertain bounds. Indeed, dependency bounds are *best-possible* bounds that contain the results of the computation under any possible case of dependency. For problems in which the computation involves variables with possibly strong but unknown dependency, the methods of Williamson and Berleant maintain a distinct advantage over both the pure and modified double-loop sampling methods.

Also, like pure double-loop sampling, the modified double-loop sampling method might become too computationally expensive for high-dimensional problems. Replacing the "parameter" loop with an optimizer should result in decreased computational cost due to the decreased number of function evaluations, but optimization over a high-dimensional space can itself remain costly. The most that can be claimed of the modified double-loop sampling approach is that it allows for the solution of a wider class of problems than pure double-loop sampling.

Although the modified double-loop sampling method retains some of the weaknesses of pure double-loop sampling, it also introduces an additional difficulty. Specifically, the functions to be optimized, $E = f(\boldsymbol{\Theta})$, are complex and non-linear and therefore multi-modal. This means that the optimization problems become global optimization problems. Depending on the complexity of the global optimization problem, the modified double-loop sampling method might be computationally infeasible. Although many sophisticated algorithms for solving global optimization problems have been developed (see (Pinter 1996; Horst, Pardalos et al. 2000; Hansen and Walster 2004)), for many problems with relatively few local minima, it is often sufficient to repeat the optimization from multiple starting points.

## 4. Discussion and Future Work

In this paper, we have introduced and formally described a computational design problem. The goal of this research is to develop computational strategies for propagating imprecise beliefs through design decision models. We argued that the currently available computational methods are unsatisfactory, and an alternative approach was introduced. The main purpose of this paper has been to clarify and communicate the problem, but we do not yet feel that the research question has been satisfactorily answered. Further work remains to be done. In particular, the proposed method needs to be numerically validated, the problem of global optimization needs to be addressed, methods for parameterizing more general p-boxes need to be studied, and means of accounting for unknown dependence need to be developed. While we believe that the

optimization method proposed in this paper is an improvement over the available methods for some classes of design problems, we do not yet feel that we have found a fully satisfactory solution. Hopefully, this paper will lead to development of new alternative strategies for computing with imprecise information.

**Numerical validation**

Before the advantage of using the modified double-loop sampling method can be confirmed, numerical experiments must first be completed. Essentially, two questions need to be asked: does the modified double-loop sampling method provide results that are reasonably close to the theoretical best-possible bounds, and is the modified double-loop sampling method substantially more computationally efficient that pure double-loop sampling? Some experimentation with these methods has already been carried out (Rekuc, Aughenbaugh et al. 2006), but thorough validation requires a more systematic study.

The proposed experiments will involve three stages. First, in order to assure accuracy, the modified and pure double-loop sampling methods will be compared with the best-possible bounds approaches for a simple sum of p-boxes. It needs to be shown that an arbitrarily small degree of error can be achieved with both of these methods using only a reasonable number of samples. At this stage in the experimentation, a study of the relative efficiency of the two double-loop sampling methods can be conducted. The two sampling methods will be compared in terms of the number of function evaluations required as well as the overall CPU processing time.

The second stage of the proposed numerical validation, will involve another simple algebraic model. The second model will be sufficiently more complex so as to involve several more uncertain quantities as well as repeated variables. The presence of repeated variables results in over-conservative bounds for interval arithmetic operations. Since the best-possible bound methods for propagating imprecise probabilities make use of the operations of interval arithmetic, they result in over-conservative bounds in the presence of repeated variables. This is one aspect in which the double-loop sampling methods have an advantage over the dependency bounds convolution and the distribution envelope approaches. Although Ferson has solved repeated variable problems using subinterval reconstitution within the dependency bounds convolution algorithms (Ferson and Hajagos 2004), it is unclear how efficient this method is for handling problems with a large number of uncertain inputs. The purpose of this second stage is to test the efficiency of the modified double-loop sampling method for problems with a greater number of uncertain quantities. It will also be interesting to compare the accuracy and the efficiency of the modified double-loop sampling method to the dependency bounds convolution with subinterval reconstitution methods.

The third and final stage of the numerical validation of the modified double-loop sampling method will involve applying the method towards a realistic engineering design problem. At this stage, it will be impossible to compare to the dependency bounds convolution approach because of the large number of repeated variables. The previous two stages will test the accuracy of the

modified double-loop sampling as compared to the dependency bounds approach, and the final stage will test to see if the modified double-loop sampling is computationally efficient enough to propagate uncertainty through realistic engineering design problems.

**The problem of global optimization**

As mentioned previously, some numerical experimentation has been done with the modified double-loop method. Specifically, the method was applied towards the design of a gearbox. The utility curve of the gearbox as a function of the distribution parameters turned out to be multi-modal. A quick fix to this problem was attained by using multiple starting points, and this worked well for the gearbox problem. However, the gearbox utility model is relatively simple compared to other realistic design decision problems. Therefore, it is uncertain whether or not more sophisticated global optimization algorithms will be needed for complex design decision problems. If it is the case that many design utility models have a very large number of modes, then an efficient, reliable global optimization algorithm will need to be found that is compatible with the modified double-loop sampling method.

**Parameterizing general p-boxes**

Both the pure and modified double-loop sampling methods assume that the DM's imprecise beliefs can be represented as parameterized p-boxes. As was discussed previously, parameterized p-boxes are special cases of general p-boxes. It was argued that parameterized p-boxes arise frequently in practice, but not all realistic belief states can be easily represented as parameterized p-boxes. For instance, Dempster-Shafer structures are general p-boxes that result from the methods of evidence theory (Ferson, Kreinovich et al. 2002; Ferson, Hajagos et al. 2005), but there appears to be no straightforward way in which to model a Dempster-Shafer structure as a distribution with imprecise parameters. If any version of a double-loop sampling method is to be generally applicable, a means of parameterizing more general p-boxes needs to be discovered.

**Accounting for known and unknown dependence**

One of the primary advantages of the dependency bounds approaches is that they allow for the determination of theoretical best-possible bounds on the resultant p-box under any state of dependence between the uncertain quantities. By comparison, the double-loop sampling methods assume some dependence between the uncertain quantities. This is in violation of the problem statement presented in section 2.5. If a joint distribution is known, then sampling in the "probability" loop can take into account that dependence information by simply sampling from the joint distribution. However, joint distributions are not often known for engineering design problems. It is therefore desirable to further modify the double-loop sampling method such that it provides something similar to best-possible bounds in cases of unknown dependence.

**Possibility of alternative approaches**

As was discussed in section 1.5, the purpose of this paper has been to pose a problem. While a possible solution to that problem has been presented, obstacles still remain to putting the proposed method into practice. Engineering designers desire a method that allows for reliable, efficient propagation of their imprecise beliefs through complex engineering design models. The dependency bounds convolution and distribution envelope approaches are efficient and provide best-possible bounds, but they are not currently compatible with black-box models. Additionally, these methods face the dilemma of interval arithmetic with repeated variables. The pure and modified double-loop sampling methods discussed in this paper are compatible with black-box models, but they seem to be inefficient for complex engineering design problems. It is our hope that by presenting this problem statement to the reliable engineering computing community, alternative approaches will be suggested.

## Acknowledgements

This work was supported in part by NSF grant DMI-0522116. The authors would like to thank Jason Aughenbaugh, Jay Ling, and Rich Malak for their contributions.

## References

Alefeld, G. and J. Herzberger. *Introduction to Interval Computations*. New York, Academic Press, 1983.

Aughenbaugh, J. M. and C. J. J. Paredis. The Value of Using Imprecise Probabilities in Engineering Design. *ASME DETC 2005*, Long Beach, California, USA, IDETC2005-85354, 2005.

Aughenbaugh, J. M. and C. J. J. Paredis. Why Are Intervals and Imprecision Important in Engineering Design? *Reliable Engineering Computing Workshop*, Savannah, GA, USA, 2006.

Berger, J. O. *Statistical Decision Theory and Bayesian Analysis*, Springer, 1985.

Berleant, D. Automatically Verified Reasoning with Both Intervals and Probability Density. *Interval Computations*(2): 48-70, 1993.

Berleant, D. and H. Cheng. A Software Tool for Automatically Verified Operations on Intervals and Probability Distributions. *Reliable Computing* **4**(1): 71-82, 1998.

Berleant, D. and C. Goodman-Strauss. Bounding the Results of Arithmetic Operations on Random Variables of Unknown Dependency Using Intervals. *Reliable Computing* **4**(2): 147-165, 1998.

Berleant, D., L. Xie and J. Zhang. Statool: A Tool for Distribution Envelope Determination (Denv), an Interval-Based Algorithm for Arithmetic on Random Variables. *Reliable Computing* **9**(2): 91-108, 2003.

Berleant, D. and J. Zhang. Representation and Problem Solving with Distribution Envelope Determination (Denv). *Reliability Engineering & System Safety* **85**(1-3): 153-168, 2004.

Cooke, R. The Anatomy of the Squizzel - the Role of Operational Definitions in Representing Uncertainty. *Reliability Engineering & System Safety* **85**(1-3): 313, 2004.

de Finetti, B. Foresight. Its Logical Laws, Its Subjective Sources (Translated). *Studies in Subjective Probability*. H. E. Kyburg and H. E. Smokler, E. Krieger Publishing Company, 1980.

Dubois, D. and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty*, Plenum Press, 1988.

Ferson, S. RAMAS Risk Calc. New York, Lewis Publishers, 2002.

Ferson, S. and S. Donald. Probability Bounds Analysis. *International Conference on Probabilistic Safety Assessment and Management (PSAM4)*, New York, NY, Springer-Verlag, 1998.

Ferson, S. and L. Ginzburg. Different Methods Are Needed to Propagate Ignorance and Variability. *Reliability Engineering and System Safety* **54**(2-3), 1996.

Ferson, S., L. Ginzburg, V. Kreinovich, H. T. Nguyen and S. A. Starks. Uncertainty in Risk Analysis: Towards a General Second-Order Approach Combining Interval, Probabilistic, and Fuzzy Techniques. *2002 IEEE International Conference on Fuzzy Systems: FUZZ-IEEE'02, May 12-17 2002*, Honolulu, HI, United States, Institute of Electrical and Electronics Engineers Inc., 2002.

Ferson, S. and L. R. Ginzburg. Different Methods Are Needed to Propagate Ignorance and Variability. *Reliability Engineering & System Safety* **54**(2-3): 133-144, 1996.

Ferson, S. and L. R. Ginzburg. Different Methods Are Needed to Propagate Ignorance and Variability [in Risk Analysis]. *Reliability Engineering & System Safety* **54**(2-3): 133-44, 1996.

Ferson, S. and J. Hajagos. Arithmetic with Uncertain Numbers: Rigorous and (Often) Best Possible Answers. *Reliability Engineering & System Safety* **85**(1-3): 135-152, 2004.

Ferson, S., J. Hajagos, D. S. Myers and W. T. Tucker. Constructor: Synthesizing Information About Uncertain Variables, Applied Biomathematics for Sandia National Laboratories, 2005.

Ferson, S., V. Kreinovich, L. Ginzburg, D. S. Myers and K. Sentz. Constructing Probability Boxes and Dempster-Shafer Structures. Albuquerque, New Mexico, Sandia National Laboratories, 2002.

Ferson, S., R. B. Nelsen, J. Hajagos, D. Berleant, J. Zhang, W. T. Tucker, L. Ginzburg and W. L. Oberkampf. Dependence in Probabilistic Modeling, Dempster-Shafer Theory, and Probability Bounds Analysis. Albuquerque, Sandia National Laboratories, 2004.

Fishman, G. S. *Monte Carlo: Concepts, Algorithms, and Applications*. New York, Springer, 1996.

Hansen, E. and G. W. Walster. *Global Optimization Using Interval Analysis*. New York, Marcel Dekker, Inc., 2004.

Hazelrigg, G. A. A Framework for Decision-Based Design. *Journal of Mechanical Design* **120**(4): 653-658, 1998.

Helton, J. C. and F. J. Davis. Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems. *Reliability Engineering & System Safety* **81**: 23-69, 2003.

Hofer, E. When to Separate Uncertainties and When Not to Separate. *Reliability Engineering & System Safety* **54**(2-3): 113-118, 1996.

Hofer, E., M. Kloos, B. Krzykacz-Hausmann, J. Peschke and M. Woltereck. An Approximate Epistemic Uncertainty Analysis Approach in the Presence of Epistemic and Aleatory Uncertainties. *Reliability Engineering & System Safety* **77**(3): 229-38, 2002.

Hoffman, F. O. and J. S. Hammonds. Propagation of Uncertainty in Risk Assessments: The Need to Distinguish between Uncertainty Due to Lack of Knowledge and Uncertainty Due to Variability. *Risk Analysis* **14**(5): 707-712, 1994.

Horst, R., P. M. Pardalos and N. V. Thoai. *Introduction to Global Optimization*. Dordrecht, Kluwer, 2000.

Kearfott, R. B. and V. Kreinovich. *Applications of Interval Computations*. AH Dordrecht, The Netherlands, Kluwer Academic Publishers, 1996.

Kreinovich, V. and S. Ferson. A New Cauchy-Based Black-Box Technique for Uncertainty in Risk Analysis. *Reliability Engineering & System Safety* **85**: 267-279, 2004.

Kreinovich, V., S. Ferson, L. Ginzburg, H. Schulte, M. Barry and H. Nguyen. From Interval Methods of Representing Uncertainty to a General Description of Uncertainty. *International Conference on Information Technology*, Bhubaneswar, India, McGraw-Hill, 1999.

Law, A. M. and D. W. Kelton. *Simulation Modeling and Analysis*. St. Louis, McGraw-Hill, 2000.

Levi, I. *The Enterprise of Knowledge, an Essay on Knowledge, Credal Probability, and Chance*. Cambridge, MA, MIT Press, 1980.

Marston, M., J. K. Allen and F. Mistree. The Decision Support Problem Technique: Integrating Descriptive and Normative Approaches in Decision Based Design. *Engineering Valuation and Cost Analysis* **3**: 107-129, 2000.

Mistree, F., W. F. Smith, B. A. Bras, J. K. Allen and D. Muster. Decision-Based Design. A Contemporary Paradigm for Ship Design. *1990 SNAME Annual Meeting, Oct 31-Nov 3 1990*, San Francisco, CA, USA, Publ by Soc of Naval Architects & Marine Engineers, Jersey City, NJ, USA, 1990.

Moore, R. *Methods and Applications of Interval Analysis*. Philadelphia, Society for Industrial and Applied Mathematics, 1979.

Muhanna, R. L. and R. L. Mullen. Interval Methods for Reliable Computing. *Engineering Design Reliability Handbook*. E. Nikolaidis, D. M. Ghiocel and S. Singhal. New York, NY, CRC Press**:** 12.1-12.24, 2004.

Parry, G. W. The Characterization of Uncertainty in Probabilistic Risk Assessment of Complex Systems. *Reliability Engineering and System Safety* **54**(2-3): 119-126, 1996.

Pinter, J. D. *Global Optimization in Action*. Dordrecht, Kluwer Academic Publishers, 1996.

Regan, H. M., S. Ferson and D. Berleant. Equivalence of Methods for Uncertainty Propagation of Real-Valued Random Variables. *International Journal of Approximate Reasoning* **36**(1): 1-30, 2004.

Rekuc, S. J. *Eliminating Design Alternatives under Interval-Based Uncertainty*. Thesis. Master of Science, Mechanical Engineering, Georgia Institute of Technology, 2005.

Rekuc, S. J., J. M. Aughenbaugh, M. Bruns and C. J. J. Paredis. Eliminating Design Alternatives Based on Imprecise Information. *Society of Automotive Engineering World Congress*, Detroit, Michigan, 06M-269, 2006.

Savage, L. J. *The Foundation of Statistics*. New York, Dover Publications, 1972.

Sobek, D. K., A. Ward and J. K. Liker. Toyota's Principles of Set-Based Concurrent Engineering. *Sloan Management Review*, 1999.

Sobek, D. K. I. and A. C. Ward. Principles from Toyota's Set-Based Concurrent Engineering Process. *ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, Irvine, California, 1996.

Springer, M. D. *The Algebra of Random Variables*. New York, John Wiley & Sons, 1979.

Trejo, R. and V. Kreinovich. Error Estimations for Indirect Measurements: Randomized Vs. Deterministic Algorithms for `Black-Box' Programs. *Handbook on Randomized Computing*. S. Rajasekaran, P. Pardalos, J. Reif and J. Rolim, Kluwer**:** 673-729, 2001.

Troffaes, M. C. M. Decision Making with Imprecise Probabilities: A Short Review. *The Society for Imprecise Probability Theory and Applications*. **2:** 4-7, 2004.

Walley, P. *Statistical Reasoning with Imprecise Probabilities*. London, Chapman and Hall, 1991.

Williamson, R. C. *Probabilistic Arithmetic*. Thesis., Electrical Engineering, University of Queensland, 1989.

Williamson, R. C. and T. Downs. Probabilistic Arithmetic I: Numerical Methods for Calculating Convolutions and Dependency Bounds". *International Journal of Approximate Reasoning* **4**: 89-158, 1990.

Winkler, R. L. Uncertainty in Probabilistic Risk Assessment. *Reliability Engineering & System Safety* **54**(2-3): 127-132, 1996.

Yager, R. R. Arithmetic and Other Operations on Dempster-Shafer Structures. *International Journal of Man-Machine Studies* **25**: 357-366, 1986.

Zadeh, L. A. Fuzzy Sets. *Information and Control* **8**: 338-353, 1965.