Optimal Elimination of Inconsistency in Expert Knowledge: Formulation of the Problem, Fast Algorithms

Timothy J. Ross
Civil Engineering Dept.
University of New Mexico
Albuquerque, NM 87131, USA
ross@unm.edu

Berlin Wu
Mathematics Dept.
National Chengchi University
Taipei, Taiwan
berlin@math.nccu.edu.tw

Vladik Kreinovich
Dept. of Computer Science
University of Texas
El Paso, TX 79968, USA
vladik@cs.utep.edu

Abstract

Expert knowledge is sometimes inconsistent. In this paper, we describe the problem of eliminating this inconsistency as an optimization problem, and present fast algorithms for solving this problem.

Keywords: expert knowledge, inconsistency, optimization

1 Introduction

1.1 Expert Knowledge Is Important

In many real-life situations, we have only a few high-level experts who can make high-quality decisions. It is desirable to incorporate their knowledge into an automated system which would help make similar good quality decisions when these top experts are not available. To design such systems, we must elicit the knowledge from the experts.

1.2 Expert Knowledge Is Sometimes Inconsistent

This elicitation problem is difficult because a large part of this expert knowledge is imprecise. When the experts articulate this informal knowledge in computer-understandable formal terms, they inevitably somewhat distort their knowledge. As a result, the expert knowledge which was perfectly consistent may result in somewhat inconsistent formal statements:

- It could be that there is a contradiction between the statements presented by different experts.
- It could also be that several statements given by the same expert are inconsistent.

1.3 Experts Sometimes Cannot Easily Handle This Inconsistency, So We Need an Automated Inconsistency-Elimination Tool

Ideally, if we encounter such an inconsistency, we would like to point out this contradiction to the experts and ask them to resolve it before we input the inconsistent data into the computer-based system. However, these inconsistencies reflect not internal inconsistencies of the expert's reasoning, but rather (inevitable) differences between the expert's internal reasoning and its formalized representation. As a result, experts themselves often cannot provide us with any meaningful guidance about the best way to resolve these inconsistencies. Our experience shows that, if faced with such an inconsistency, experts often make arbitrary changes in their original estimates, changes which are kind of "random" in the sense that in a similar situation the same expert can make a completely different adjustment.

Since we cannot rely on experts to provide us with a meaningful and "consistent" way of eliminating inconsistency, we must therefore design automatic inconsistency-elimination tools and use them as a part of knowledge elicitation.

1.4 What We Are Planning to Do

In this paper, we formulate the problem of optimal inconsistency elimination, and provide fast algorithms for such elimination for the practically important situation when the expert knowledge is about the numerical value of a physical characteristic (e.g., geometric and mechanical parameters of a complex structure). We will consider two cases:

- when the inconsistency is between statements made by different experts, and
- when the inconsistency is between different statements made by the same expert.

2 Eliminating Inconsistency Between Statements Made By Different Experts

2.1 Expert Estimates: General Formulation

Let n denote the total number of experts who estimate the value of a given physical characteristic (such as width, density, etc.), and let $\mathbf{x}_i = [x_i^-, x_i^+]$ be an interval estimate provided by i-th expert ($i = 1, 2, \ldots, n$). In other words, expert i claims that the actual (unknown) value x of the estimated physical characteristic is in the interval $[x_i^-, x_i^+]$.

2.2 If Experts Are Consistent With Each Other, the Situation Is Easy to Handle

If all the experts are correct, then the actual value of x belongs to all n intervals. The resulting set of possible values of x is thus equal to the intersection

$$\mathbf{x} = \mathbf{x}_1 \cap \mathbf{x}_2 \cap \ldots \cap \mathbf{x}_n$$

of these n intervals. This intersection $\mathbf{x} = [x^-, x^+]$ is easy to compute: $x^- = \max(x_1^-, \ldots, x_n^-)$ and $x^+ = \min(x_1^+, \ldots, x_n^+)$.

2.3 The Problem: Experts' Statements Can Be Inconsistent With Each Other

The problem emerges when the intersection of n intervals is empty, i.e., when $\max(x_1^-, \ldots, x_n^-) > \min(x_1^+, \ldots, x_n^+)$. In this case, no real number x can satisfy all the experts' requirements $x \in [x_i^-, x_i^+]$ and therefore, some of these requirements are false. Thus, to design a consistent expert system, we must dismiss some of the expert estimates before computing the intersection.

2.4 We Want To Find the Optimal (Most Probable) Way of Eliminating Inconsistency

There are many ways of eliminating inconsistency. For example, we can always get a consistent statement by simply dismissing all but one interval. From the purely mathematical viewpoint, it makes sense: we get a single interval, and this single interval is, of course, consistent. However, from the viewpoint of our problem, what we are doing is, in effect, proclaiming that $n-1$ experts are wrong. It can be that one of the experts is wrong, it can even happen that two experts are wrong, but it is hardly probable that all but one of them are wrong – if this is true what kind of experts are they? So, if we simply dismiss all expert estimates but one, we get a consistent set, but this consistent set is very improbable.

Thus, we do not simply want to eliminate inconsistency, we want to select the *optimal* ("most probable") way of eliminating inconsistency.

So far, we have used the word "most probable" in the informal sense. To formalize this problem, we must describe the corresponding probabilities.

2.5 Possible Additional Knowledge About Reliability of Different Experts

Depending on our knowledge about the experts, we will consider two different formulations of this problem:

- We may already have some experience with this group of experts, and so we know, from experience, for each expert i, the fraction p_i of cases in which this expert turned out to be correct. In our words, we know the *probability* p_i that the i-th expert's statement is correct.
- It may also happen that we do not have any preliminary experience with this particular group of experts, so we have no reason to assume that one of them is more reliable than the others. In this case, it makes sense to assume that all experts have the same (unknown) probability p of making a correct statement.

  > It is safe to assume that since we consider all these experts to be really experts, the probability p that an expert is correct should be larger than the probability $1-p$ that the expert is wrong, i.e., $p > 1-p$ (which is equivalent to $p > 0.5$).

From the mathematical viewpoint, the second situation can be considered as a particular case of the first one, when $p_1 = \ldots = p_n = p$; the only difference is that in the first case, we know the probabilities p_i, while in the second case, we do not know them (we only know that they are equal to each other).

2.6 Experts Are Independent

In this paper, we will assume that the experts are independent. This assumption makes sense: e.g., if the opinions of the two experts are highly correlated, then there is no sense to interview both of them: the opinion of the first expert describes, with a high accuracy, the opinion of the second expert as well. Thus, e.g.:

- the probability that the experts i and j are both correct is equal to $p_i \cdot p_j$;
- the probability that the expert i is correct and the expert j is not correct is equal to $p_i \cdot (1 - p_j)$.

2.7 Towards the Mathematical Formulation of the Problem

Our goal is to select a subset $S \subset \{1, 2, \ldots, n\}$ for which the corresponding statements are consistent, i.e., for which

$$\bigcap_{i \in S} \mathbf{x}_i \neq \emptyset. \qquad (1)$$

For each such selection S, its probability $p(S)$ is equal to:

$$p(S) = \left(\prod_{i \in S} p_i\right) \cdot \left(\prod_{i \notin S} (1 - p_i)\right). \qquad (2)$$

We must therefore, find a set S for which $p(S) \to \max$ among all sets S which satisfy the condition (1).

2.8 In Principle, We Can Use Exhaustive Search, But This Is Intractable

There are only finitely many sets $S \subset \{1, \ldots, n\}$, so in principle, we could simply test them all and find the one for which the probability $p(S)$ is the largest possible. Unfortunately, this "exhaustive search" strategy is very time-consuming: indeed, a set of n elements has 2^n subsets, so for a realistic case of $n = 30$ experts, we need to test a billion of different subsets. For even larger number of experts, this search becomes completely intractable. We therefore need a more sophisticated algorithm for selecting the most probable combination of intervals.

This algorithm will be presented now.

2.9 Fast Algorithm for Solving This Problem (Case When We Know p_i)

The proposed algorithm consists of the following stages:

1. First, we sort all $2n$ endpoints

   $$x_1^-, \ldots, x_n^-, x_1^+, \ldots, x_n^+$$

   into an ordered sequence of real numbers

   $$x_{(1)} \leq x_{(k)} \leq \ldots \leq x_{(2n)}.$$

   This sorting requires $O(n \cdot \log(n))$ elementary computation steps (see, e.g., [3]).

2. Then, for each $k = 1, \ldots, 2n - 1$, we consider the interval $[x_{(k)}, x_{(k+1)}]$. For this interval, we check, for every $i = 1, \ldots, n$, whether this interval belongs to $\mathbf{x}_i$ or not. Thus, we form the set $S_{(k)}$ of all indices for which $[x_{(k)}, x_{(k+1)}] \subseteq \mathbf{x}_i$, and and then compute the corresponding value $p(S_{(k)})$ by using the formula (2).

   Each checking requires $2n$ comparisons, and computing $p(S)$ requires n multiplications. So, for each k, we need $\leq 3n$ steps, and the total number of computational steps required to perform this procedure for all $2n-1$ values of k is equal to $O(n) \cdot O(n) = O(n^2)$.

3. As a result of the second stage, we get $2n - 1$ different values of $p(S_{(k)})$. To find the largest value (or values), we compare them with each other: we start with the value corresponding to the leftmost interval $[x_{(1)}, x_{(2)}]$, mark it as the largest-so-far, and then go over other values, replacing the largest-so-far with the next one if the next value of $p(S_{(k)})$ is indeed larger.

   For each of $2n - 1$ values, we perform a single comparisons, so this stage requires $O(n)$ elementary computational steps.

The total number of steps is therefore equal to

$$O(n \cdot \log(n)) + O(n^2) + O(n) = O(n^2),$$

i.e., we get a quadratic-time algorithm for finding the optimal (most probable) consistent combination of expert statements.

2.10 Pedagogical Example

Let us illustrate the above algorithm on the following simple example. Let

$$\mathbf{x}_1 = [0.0, 1.0], \ \ \mathbf{x}_2 = [0.5, 1.5],$$

$$\mathbf{x}_3 = [0.7, 1.2], \ \ \mathbf{x}_4 = [1.3, 1.6].$$

In this case,

$$\max(x_1^-, \ldots, x_4^-) = 1.3 > \min(x_1^+, \ldots, x_4^+) = 1.0$$

and thus, the intersection of all four intervals is empty.

Let us assume that $p_1 = p_2 = p_3 = 0.7$ and $p_4 = 0.8$. Then, the algorithm results in the following selection:

1. First, we sort all 8 values $x_i^\pm$ into a sequence

   $$x_{(1)} = 0.0 \leq x_{(2)} = 0.5 \leq x_{(3)} = 0.7 \leq$$

   $$x_{(4)} = 1.0 \leq x_{(5)} = 1.2 \leq x_{(6)} = 1.3 \leq$$

   $$x_{(7)} = 1.5 \leq x_{(8)} = 1.6.$$

2. Second, for each of 7 intervals $[x_{(k)}, x_{(k+1)}]$, we compute the corresponding value $p(S_{(k)})$:

   - For $k = 1$, we have $[x_{(1)}, x_{(2)}] = [0.0, 0.5]$, so $S_{(1)} = \{1\}$, and

     $$p(S_{(1)}) = 0.7 \cdot 0.3 \cdot 0.3 \cdot 0.2 = 0.0126.$$

   - For $k = 2$, we have $[x_{(2)}, x_{(3)}] = [0.5, 0.7]$, so $S_{(2)} = \{1, 2\}$, and

     $$p(S_{(2)}) = 0.7 \cdot 0.7 \cdot 0.3 \cdot 0.2 = 0.0294.$$

   - For $k = 3$, we have $[x_{(3)}, x_{(4)}] = [0.7, 1.0]$, so $S_{(3)} = \{1, 2, 3\}$, and

     $$p(S_{(3)}) = 0.7 \cdot 0.7 \cdot 0.7 \cdot 0.2 = 0.0686.$$

   - For $k = 4$, we have $[x_{(4)}, x_{(5)}] = [1.0, 1.2]$, so $S_{(4)} = \{2, 3\}$, and

     $$p(S_{(4)}) = 0.3 \cdot 0.7 \cdot 0.7 \cdot 0.2 = 0.0294.$$

   - For $k = 5$, we have $[x_{(5)}, x_{(6)}] = [1.2, 1.3]$, so $S_{(1)} = \{2\}$, and

     $$p(S_{(5)}) = 0.3 \cdot 0.7 \cdot 0.3 \cdot 0.2 = 0.0126.$$

   - For $k = 6$, we have $[x_{(6)}, x_{(7)}] = [1.3, 1.5]$, so $S_{(1)} = \{2, 4\}$, and

     $$p(S_{(6)}) = 0.3 \cdot 0.7 \cdot 0.3 \cdot 0.8 = 0.0504.$$

   - For $k = 7$, we have $[x_{(7)}, x_{(8)}] = [1.5, 1.6]$, so $S_{(1)} = \{4\}$, and

     $$p(S_{(1)}) = 0.3 \cdot 0.3 \cdot 0.3 \cdot 0.8 = 0.0202.$$

3. Among these 7 values, the largest corresponds to $S_{(3)} = \{1, 2, 3\}$, so we dismiss the estimate of the 4th expert.

2.11 How To Modify the Above Algorithm If We Do Not Know the Probabilities p_i

If we do not know the probabilities p_i (i.e., if we only know that $p_i = p$ for some (unknown) value $p > 0.5$), then we cannot implement the second stage of the above algorithm – because we cannot compute the value $p(S)$.

However, there is a way around it: the only places where we use the values $p(S)$ is on the third stage of this algorithm, and on this stage, we do not need the actual *values* of $p(S)$, we only need to know which values are larger and which are smaller. For the case when $p_i = p$ for all i, the probability (2) turns into

$$p(S) = p^{|S|} \cdot (1-p)^{n-|S|},$$

where $|S|$ denotes the number of elements in the set S. This expression can be rewritten as

$$p(S) = \left(\frac{p}{1-p}\right)^{|S|} \cdot (1-p)^n.$$

Since $p > 1 - p$, we can conclude that the more elements S has, the larger the corresponding value of $p(S)$. Thus, instead of computing $p(S)$, we can simply compute $|S|$ and then select a consistent set S which contains the largest possible number of intervals.

In other words, we dismiss the smallest possible number of expert estimates: this makes perfect sense.

Thus, we arrive at the following algorithm:

2.12 Fast Algorithm for Solving This Problem (Case When We Do Not Know the Probabilities p_i)

For this case, the proposed algorithm consists of the following stages:

1. First, we sort all $2n$ endpoints

   $$x_1^-, \ldots, x_n^-, x_1^+, \ldots, x_n^+$$

   into an ordered sequence of real numbers

   $$x_{(1)} \le x_{(k)} \le \ldots \le x_{(2n)}.$$

2. Then, for each $k = 1, \ldots, 2n-1$, we consider the interval $[x_{(k)}, x_{(k+1)}]$. For this interval, we check, for every $i = 1, \ldots, n$, whether this interval belongs to $\mathbf{x}_i$ or not. Thus, we form the set $S_{(k)}$ of all indices for which $[x_{(k)}, x_{(k+1)}] \subseteq \mathbf{x}_i$, and then compute the total number $|S_{(k)}|$ of elements in this set.

3. As a result of the second stage, we get $2n-1$ different values of $|S_{(k)}|$. To find the largest value (or values), we compare them with each other: we start with the value corresponding to the leftmost interval $[x_{(1)}, x_{(2)}]$, mark it as the largest-so-far, and then go over other values, replacing the largest-so-far with the next one if the next value of $|S_{(k)}|$ is indeed larger.

This version also requires $O(n^2)$ elementary computation steps.

2.13 Pedagogical Example

If in the above example

$$\mathbf{x}_1 = [0.0, 1.0], \ \ \mathbf{x}_2 = [0.5, 1.5],$$

$$\mathbf{x}_3 = [0.7, 1.2], \ \ \mathbf{x}_4 = [1.3, 1.6],$$

we do not know the probabilities, then, on the second stage, we compute the values

$$|S_{(1)}| = 1,\ \ |S_{(2)}| = 2,\ \ |S_{(3)}| = 3,\ \ |S_{(4)}| = 2,$$
$$|S_{(5)}| = 1,\ \ |S_{(6)}| = 2,\ \ |S_{(7)}| = 1,$$

and select the set $S_{(3)}$ for which this number is the largest.

3 Eliminating Inconsistency Between Different Statements Made By the Same Expert

3.1 Nested Intervals (Fuzzy Set) Instead of a Single Interval

In the above description, we assumed that for each physical characteristic x, each expert provides a single interval $[x^-, x^+]$ of possible values of x. In reality, in addition to the interval about which the expert is absolutely sure that x is there, the expert may provide narrower intervals corresponding to smaller degrees of confidence. For example, an expert may be absolutely sure that $x \in [0.0, 1.0]$, almost absolutely sure that $x \in [0.2, 0.8]$, somewhat sure that $x \in [0.4, 0.7]$, etc.

As a result, we get a "nested" ("consonant") collection of intervals $\mathbf{x}_1 \subseteq \mathbf{x}_2 \subseteq \ldots \subseteq \mathbf{x}_n$ which correspond to different degrees of uncertainty. Such a nested collection of intervals forms a *fuzzy set* [1, 5, 6, 7], in which different intervals can be viewed as α-cuts.

3.2 Some Intervals May Be Erroneous, So We Must Choose a Consonant Sub-Collection

In the ideal situation, the intervals are nested. In real life, some intervals provided by an expert may not be ordered by inclusion. In this case, we must select a consonant subset.

Let us formulate this problem in more precise terms.

Definition 1. *We say that two intervals* $\mathbf{x}$ *and* $\mathbf{y}$ *are compatible if either* $\mathbf{x} \subseteq \mathbf{y}$, *or* $\mathbf{y} \subseteq \mathbf{x}$.

Definition 2. *We say that a collection of intervals is consonant if every two intervals from this collection are compatible.*

In these terms, our problem can be formulated as follows:

- we have a collection of intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ which is not consonant;
- we must select a consonant sub-collection, i.e., a set $S \subset \{1, 2, \ldots, n\}$ for which the intervals $\mathbf{x}_i$, $i \in S$ are consonant.

We would like to choose the "optimal" ("most probable") sub-collection.

3.3 We Want To Find the Optimal (Most Probable) Way of Eliminating Inconsistency

We know that some intervals in the original collection can be erroneous. Since we have no information about these probabilities, we can assume that each of these intervals has the same (unknown) probability p to be erroneous. The probability corresponding to each sub-collection can be therefore described by the formula $p(S) = p^{|S|} \cdot (1-p)^{n-|S|}$. We know, from the first problem, that this value is the largest if and only if S contains the largest number of elements. So, the problem of selecting the optimal consonant sub-collection can be reformulated as follows:

- we have a collection of intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ which is not consonant;
- we must select a consonant sub-collection S which contains the largest possible number of intervals.

(In other words, we want to dismiss the smallest possible number of intervals – which makes perfect sense.)

3.4 In Principle, We Can Use Exhaustive Search, But This Is Intractable

Similarly to the first problem, we can find the solution to this one by exhaustive search, but for large n, this is intractable, so we need a faster algorithm. This algorithm is described below.

3.5 Fast Algorithm for Solving This Problem

We start with a collection of n intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$.

- First, for every two intervals $\mathbf{x}_i$ and $\mathbf{x}_j$ from the original collection, we check whether $\mathbf{x}_i \subseteq \mathbf{x}_j$, and put the result ("true" or "false") into the corresponding cell r_{ij} of a special $n \times n$ table.

  For n intervals, this stage requires $2n^2 = O(n^2)$ comparisons.

- For each interval $\mathbf{x}_i$, we check whether this interval is minimal, i.e., whether $r_{ji} =$"false" for all j. Then, we mark all the intervals which satisfy this property as intervals of Level 1.
- Then, for every interval which has not yet been marked, we check whether this interval is minimal among un-marked ones, i.e., whether $r_{ji} =$"false" for all un-marked intervals $\mathbf{x}_j$. For each interval which satisfies this property, we find all its marked sub-intervals (i.e., all marked intervals $\mathbf{x}_j$ for which $r_{ji} =$"true"); among these sub-intervals, we find the ones with the largest level ℓ. Then, we mark $\mathbf{x}_i$ as an interval of level $\ell + 1$, and list nearby all its sub-intervals of level ℓ.

- We then repeat this procedure until all intervals are marked.

- At the end, we select the desired consonant sub-collection $\mathbf{y}_1, \ldots, \mathbf{y}_k$:

  - as $\mathbf{y}_1$, we select the interval with the highest level L;

  - after $\mathbf{y}_i$ is selected (on level $L - i + 1$), and $i < L$, we select, as $\mathbf{y}_{i+1}$, any of the $\mathbf{y}_i$'s sub-intervals of level $L - i$.

On each stage, at least one interval is marked, so after at most n stages, the procedure will stop. On each stage, for each of n intervals, we check n elements of the table; thus, each stage requires $O(n^2)$ steps, and n stages require $O(n \cdot n^2) = O(n^3)$ steps. The algorithm as a whole requires $O(n^2) + O(n^3) = O(n^3)$ steps. Thus, this algorithm requires cubic time.

3.6 Pedagogical Example

Let

$$\mathbf{x}_1 = [0.0, 1.0], \;\; \mathbf{x}_2 = [0.4, 0.7],$$

$$\mathbf{x}_3 = [0.2, 0.8], \;\; \mathbf{x}_4 = [0.9, 1.0].$$

This collection is not consonant, because, e.g., neither $\mathbf{x}_3 \subseteq \mathbf{x}_4$, nor $\mathbf{x}_4 \subseteq \mathbf{x}_3$.

- According to the above algorithm, first, we pick intervals which do not contain any other intervals. In our original collection, there are two such intervals: $\mathbf{x}_2 = [0.4, 0.7]$ and $\mathbf{x}_4 = [0.9, 1.0]$. These intervals are of Level 1.

- On the next stage, we compare two remaining (unmarked) intervals $\mathbf{x}_1$ and $\mathbf{x}_3$. The only minimal one among them is $\mathbf{x}_3$. Its only marked sub-interval is $\mathbf{x}_2$ of Level 1; thus, we assign to $\mathbf{x}_3$ Level 2, and list $\mathbf{x}_2$ as its sub-interval of Level 1.

- There is only one remaining un-marked interval $\mathbf{x}_1$, so this un-marked interval is therefore minimal. It has three marked sub-intervals: $\mathbf{x}_3$ of Level 2, and $\mathbf{x}_2$ and $\mathbf{x}_4$ of Level 1. Thus, the largest level of sub-intervals is 2, so we mark the interval $\mathbf{x}_1$ as Level 3, and list $\mathbf{x}_3$ as its sub-interval of Level 2.

- All intervals are marked now, so we can start forming the desired consonant sub-collection:

  - First, we pick the interval $\mathbf{x}_1 = [0.0, 1.0]$ with the highest level as the first element $\mathbf{y}_1$ of the desired collection. Here, $L = 3$.

  - As $\mathbf{y}_2$, we pick $\mathbf{y}_1$'s sub-interval $\mathbf{x}_3 = [0.2, 0.8]$ of level $L - 1 = 2$.

  - As $\mathbf{y}_3$, we pick $\mathbf{y}_2$'s sub-interval $\mathbf{x}_2 = [0.4, 0.7]$ of level $L - 2 = 1$.

This means that we dismiss the interval $\mathbf{x}_4$.

4 An Application of the Above Algorithm to Error Estimation

4.1 Formulation of the Problem

In this section, we show that although the above method was originally designed to deal with inconsistency of expert knowledge, it can also be used in error estimation, when expert knowledge is only indirectly present. Specifically, we have a simulation model of a complex system. The results of the simulation depend on the values of certain parameters of the system, parameters whose values are not exactly known. For example, for a mechanical structure, these parameters include geometric parameters such as widths, heights, etc., and mechanical parameters such as density, Young's modulus, etc. For each of these parameters $p_1, \ldots, p_n$, we know the "average" (nominal) value $p_i^{(0)}$, and we know the interval $[p_i^-, p_i^+]$ of possible values of this parameter.

We are interested in a certain characteristic x of the system. For example, for a mechanical structure, we may be interested in this structure's stability – measured, e.g., by the maximal load that this structure can withstand. We want to make sure that this characteristic indeed exceeds the desired threshold x_0 (e.g., that the structure can withstand the given load x_0).

For each combination of values $p_1 \in [p_1^-, p_1^+]$, ..., $p_n \in [p_n^-, p_n^+]$, the simulation model enables us to compute the value $x = f(p_1, \ldots, p_n)$ of the desired characteristic. In particular, for the nominal values $p_i = p_i^{(0)}$ of the parameters, we get the value $x^{(0)} = f(p_1^{(0)}, \ldots, p_n^{(0)})$. Since we do not know what exactly are the actual values p_i, it is not enough to check that this nominal value is OK (i.e., that $x^{(0)} \geq x_0$). We must also test this condition under different combinations of the parameters p_i. The number of possible combinations grows exponentially with n. Thus, for a realistically large number of parameters, an exhaustive-search type checking all possible combinations becomes an intractable problem.

We need a faster algorithm.

4.2 Conservative Estimate (Linearization): Idea

A traditional engineering approach to this problem is to *linearize* it. The main idea of this approach is as follows. Usually, in most engineering problems, all the dependencies are smooth. In particular, it is safe to assume that the dependence $f(p_1,\ldots,p_n)$ of the desired characteristic x on the parameters p_i is also smooth. To utilize this smoothness, we can take into consideration the fact that usually, the intervals $[p_i^-,p_i^+]$ are reasonably narrow, and thus, we can describe p_i as $p_i=p^{(0)}+\Delta p_i$, where $\Delta p_i\in[p_i^- - p_i^{(0)}, p_i^+ - p_i^{(o)}]$ is reasonably small. Thus, we can expand the dependence

$$f(p_1,\ldots,p_n)=f(p_1^{(0)}+\Delta p_1,\ldots,p_n^{(0)}+\Delta p_n)$$

in a Taylor series in Δp_i and ignore quadratic and higher order terms in this expansion. As a result, we get the following approximate expression for $f(p_1,\ldots,p_n)$:

$$f(p_1,\ldots,p_n)=x^{(0)}+c_1\cdot\Delta p_1+\ldots+c_n\cdot\Delta p_n, \quad (3)$$

where we denoted

$$c_i=\frac{\partial f}{\partial p_i}_{|p_i=p_i^{(0)}}. \quad (4)$$

If we know the derivatives c_i, then we can find the smallest and the largest values of the expression (3): namely, the sum attains the largest possible value if each term attains the largest possible value, and depending on the sign of c_i, this largest possible value if attained when Δp_i reaches either the lower endpoint or the upper endpoint of its interval of possible values.

Often, possible bounds on p_i are described in terms of a largest possible distance Δ_i between the actual value p_i and the nominal value $p_i^{(0)}$. In this case, $p_i^-=p^{(0)}-\Delta_i$ and $p_i^+=p^{(0)}+\Delta_i$. Then, the largest possible value of $c_i\cdot\Delta p_i$ is equal to $|c_i|\cdot\Delta_i$, the smallest possible value of $c_i\cdot\Delta p_i$ is equal to $-|c_i|\cdot\Delta_i$, and therefore, the linear combination (3) can take arbitrary values from the interval $[x^-,x^+]=[x^{(0)}-\Delta,x^{(0)}+\Delta]$, where we denoted

$$\Delta\stackrel{\text{def}}{=}|c_1|\cdot\Delta_1+\ldots+|c_n|\cdot\Delta_n.$$

In general, if the function f was given by an analytical expression, then we would be able to explicitly differentiate it and get the desired coefficients c_i. In reality, the function f is not known analytically, it is only given as a complex program, so we need to use numerical differentiation to find the values. For example, if we take $p_2=p_2^{(0)},\ldots,p_n=p_n^{(0)}$ (i.e., $\Delta p_2=\ldots=\Delta p_n=0$), then (3) turns into

$$f(p_1,p_2^{(0)},\ldots,p_n^{(0)})=x^{(0)}+c_1\cdot\Delta p_1. \quad (5)$$

This expression is a linear function of Δp_1. It is known that a linear function attains its largest and smallest values on each interval at the endpoints of this interval. Thus, the largest and the smallest values of the expression (5) are attained when Δp_1 is equal to the endpoints of the corresponding interval (and thus, when p_1 is equal to either p_1^- or to p_1^+). Hence, to find the smallest and the largest possible values of (5), we can compute $f(p_1^-,p_2^{(0)},\ldots,p_n^{(0)})$ and $f(p_1^+,p_2^{(0)},\ldots,p_n^{(0)})$.

We will denote the smallest and the largest of these values by x_1^- and x_1^+. In these notations, the combination $c_0+c_1\cdot\Delta p_1$ takes arbitrary values from the interval $[x_1^-,x_1^+]$. Hence, the product $c_1\cdot\Delta p_1$ can take arbitrary values from the interval $[x_1^- - x^{(0)}, x_1^+ - x^{(0)}]$.

Similarly, for every i, the product $c_i\cdot\Delta p_i$ can take arbitrary values from the interval $[x_i^- - x^{(0)}, x_i^+ - x^{(0)}]$. Therefore, the linear combination (3) can take arbitrary values from the interval $[x^-,x^+]$, where

$$x^-=x^{(0)}+(x_1^- - x^{(0)})+\ldots+(x_n^- - x^{(0)}); \quad (6)$$

$$x^+=x^{(0)}+(x_1^+ - x^{(0)})+\ldots+(x_n^+ - x^{(0)}). \quad (7)$$

4.3 Conservative Estimate (Linearization): Algorithm

We know the algorithm $f(p_1,\ldots,p_n)$ which transforms the values of the parameters p_i into an estimate for the desired characteristic x. For every parameter p_i, we know its nominal value $p_i^{(0)}$, and we know the interval $[p_i^-,p_i^+]$ of possible values. We want to estimate the range of x when $p_i\in[p_i^-,p_i^+]$.

- First, we compute the nominal value $x^{(0)}$ of x by applying f to the nominal values of the parameters: $x^{(0)}\stackrel{\text{def}}{=}f(x_1^{(0)},\ldots,x_n^{(0)})$.
- Then, for every i, we change p_i to p_i^- and p_i^+ while keeping all other parameters at their nominal values, and apply the simulation function f. As a result, we get two values of x:

$$x_i^{(-)}\stackrel{\text{def}}{=}f(p_1^{(0)},\ldots,p_{i-1}^{(0)},p_i^-,p_{i+1}^{(0)},\ldots,p_n^{(0)}); \quad (8)$$

$$x_i^{(+)}\stackrel{\text{def}}{=}f(p_1^{(0)},\ldots,p_{i-1}^{(0)},p_i^+,p_{i+1}^{(0)},\ldots,p_n^{(0)}); \quad (9)$$

  we denote the smallest of these two values by x_i^-, and the largest by x_i^+.
- Finally, we compute the endpoints x^- and x^+ of the interval of possible values of x by using the formulas (6) and (7).

4.4 Conservative Estimate (Linearization): Problem

The main problem with the above conservative estimate is that it is too conservative: the worst-case upper bound (that we computed) corresponds to the case when all the parameters attain their extreme values. Theoretically, such an unfortunate worst-case combination of parameters is possible, but in reality, experts often consider such a worst-case scenario to be un-realistic.

Since the worst case is not realistic, we can probably get a narrower interval of possible values of x. In the following text, we describe two ideas for such an estimate. The second of these ideas will use the above consonant-selection algorithm.

4.5 Randomized Approach: Idea

Experts claim that the extreme combination of values p_i is "improbable". To formalize this claim, let us describe a reasonable probability distribution on the set of all possible combinations $p = (p_1, \ldots, p_n)$.

The only information that we have about each value p_i is that p_i can take any value from the interval $[p_i^-, p_i^+]$. Thus, the possible values of the vector $p = (p_1, \ldots, p_n)$ form an n-dimensional box

$$[p_1^-, p_1^+] \times \ldots \times [p_n^-, p_n^+].$$

Thus, the desired probability distribution must be located in this box. There are many possible probability distributions located in this box. According to mathematical statistics, it is reasonable to select a distribution for which the entropy is the largest possible (see, e.g., [4]). Such a distribution corresponds to the case when the variables p_i are independent, and each of them is uniformly distributed on the corresponding interval $[p_i^-, p_i^+]$. What is the corresponding distribution for the value x (as described by the formula (3))?

4.6 Randomized Approach: Derivation of the Formulas

For large n, we can apply the central limit theorem (see, e.g., [10]) and conclude that the distribution is close to Gaussian. It is known that a Gaussian distribution is uniquely determined by its mean value $\bar{x}$ and standard deviation $\sigma[x]$, so let us determine these two values.

Since the formula (3) describes x as a linear combination of the random variables p_i, the mean value $\bar{x}$ is equal to the linear combination of the corresponding means $\bar{p}_i$:

$$\bar{x} = x^{(0)} + c_1 \cdot (\bar{p}_1 - p^{(0)}) + \ldots + c_n \cdot (\bar{p}_n - p^{(0)}). \quad (10)$$

Since each variable p_i is uniformly distributed on the interval $[p_i^-, p_i^+]$, its mean value is equal to the midpoint of this interval: $\bar{p}_i = (p_i^- + p_i^+)/2$. Since we do not know the values c_i, we must compute the resulting expression indirectly. Namely, from the formulas (8) and (9), we conclude that $x_i^{(-)} = x_i^{(0)} + c_i \cdot (p_i^- - p^{(0)})$ and $x_i^{(+)} = x_i^{(0)} + c_i \cdot (p_i^+ - p^{(0)})$. Taking an arithmetic average of the left-hand sides and right-hand sides of these two equalities, we conclude that

$$\bar{x}_i = x^{(0)} + c_i \cdot (\bar{p}_i - p_i^{(0)}),$$

where we denoted

$$\bar{x}_i \stackrel{\text{def}}{=} \frac{x_i^{(-)} + x_i^{(+)}}{2}. \quad (11)$$

Therefore, $c_i \cdot (\bar{p}_i - p_i^{(0)}) = \bar{x}_i - x^{(0)}$, and from (10), we conclude that

$$\bar{x} = x^{(0)} + (\bar{x}_1 - x^{(0)}) + \ldots + (\bar{x}_n - x_n^{(0)}). \quad (12)$$

To compute the standard deviation $\sigma[x]$, we can also use the fact that the random variables p_i are independent and uniformly distributed. For a random variable p_i which is uniformly distributed on an interval $[p_i^-, p_i^+]$, the standard deviation is determined by the known formula $\sigma^2[p_i] = \frac{1}{12} \cdot |p_i^+ - p_i^-|^2$. Therefore,

$$\sigma^2[x] = \sum_{i=1}^n c_i^2 \cdot \sigma^2[p_i] = \frac{1}{12} \cdot \sum_{i=1}^n (c_i \cdot (p_i^+ - p_i^-))^2. \quad (13)$$

From the formulas (8) and (9), we can conclude that

$$c_i \cdot (p_i^+ - p_i^-) = x_i^{(+)} - x_i^{(-)},$$

therefore,

$$\sigma[x] = \frac{1}{\sqrt{12}} \cdot \sqrt{\sum_{i=1}^n (x_i^+ - x_i^-)^2}. \quad (14)$$

It is known that with a high probability, the actual value of a Gaussian random variable x cannot differ from its average by more than k standard deviations, where $k = 2$ corresponds to probability ≈ 0.95, $k = 3$ corresponds to the probability ≈ 0.999, etc. Thus, we can conclude that with a high probability, $x \in [x^-, x^+]$, where

$$x^- = \bar{x} - k \cdot \sigma[x]; \;\; x^+ = \bar{x} + \cdot\sigma[x]. \quad (15)$$

Thus, we arrive at the following algorithm:

4.7 Randomized Approach: Algorithm

We know the algorithm $f(p_1, \ldots, p_n)$ which transforms the values of the parameters p_i into an estimate for the desired characteristic x. For every parameter p_i, we know its nominal value $p_i^{(0)}$, and we know the interval $[p_i^-, p_i^+]$ of possible values. We want to estimate the range of x when $p_i \in [p_i^-, p_i^+]$.

- First, we compute the nominal value $x^{(0)}$ of x by applying f to the nominal values of the parameters: $x^{(0)} \stackrel{\text{def}}{=} f(x_1^{(0)}, \ldots, x_n^{(0)})$.

- Then, for every i, we change p_i to p_i^- and p_i^+ while keeping all other parameters at their nominal values, and apply the simulation function f. In other words, we compute the values $x_i^{(-)}$ and $x_i^{(+)}$ by using the formulas (8) and (9); we denote the smallest of these two values by x_i^-, and the largest by x_i^+.

- For each i, we compute the mean value $\bar{x}_i$ by using the formula (11), and then compute $\bar{x}$ and $\sigma[x]$ by using the formulas (12) and (14).

- Finally, we compute the endpoints x^- and x^+ of the interval of possible values of x by using the formulas (15).

4.8 Single Fault: Idea

Another reasonable idea is as follows. In reliability analysis and in fault detection, it is often assumed that in a well-functioning system:

- it is possible that one component fails, but

- it is highly improbable that two or more components fail at the same time.

Similarly, it is reasonable to assume that:

- while a single parameter p_i can drastically differ from its nominal value $p_i^{(0)}$,

- it is highly improbable that two or more parameters will take values which are drastically different from their nominal values.

Thus, in estimating possible values of x, we can safely assume that *at most one* parameter p_i differs from its nominal value, while the values of all other parameters are practically equal to their nominal values.

Since we have no a priori information on which of n parameters is more probable to differ from its nominal values, it is natural to assume that all n parameters have the same probability of deviation, which is thus equal to $1/n$.

What can we then conclude about the possible values of x?

4.9 Single Fault: Formulas

When the i-th parameter is the only one which differs from its nominal value, then, due to formula (3), possible values of x range from $x_i^{(-)}$ (which corresponds to $p_i = p_i^-$) to $x_i^{(+)}$ (which corresponds to $p_i = p_i^+$). In other words, in this case, the set of possible values of x is the interval $[x_i^-, x_i^+]$.

Thus, we can conclude that with probability $1/n$, the set of possible values coincides with $[x_1^-, x_1^+]$, with the same probability $1/n$, this set coincides with $[x_2^-, x_2^+]$, etc. This description is in line with the Dempster-Shafer formalism of possibility distributions (see, e.g., [11]), which can be used to describe, for each interval I, the *belief* $Bel(I)$ and the *plausibility* $Pl(I)$ that the actual value x is in I:

$$Bel(I) = \sum_{i:[x_i^-, x_i^+] \subseteq I} \frac{1}{n}; \;\; Pl(I) = \sum_{i:[x_i^-, x_i^+] \cap I \neq \emptyset} \frac{1}{n}.$$

4.10 Single Fault: Possible Inconsistency

If all bounds on p_i are described in terms of a largest possible distance Δ_i between the actual value p_i and the nominal value $p_i^{(0)}$, then $p_i^- = p_i^{(0)} - \Delta_i$, $p_i^+ = p_i^{(0)} + \Delta_i$, and hence,

$$[x_i^-, x_i^+] = [x_i^{(0)} - |c_i| \cdot \Delta_i, x_i^{(0)} + |c_i| \cdot \Delta_i].$$

The corresponding intervals all have the value $x^{(0)}$ as their center and therefore, they should be nested (consonant).

In real life, due to the non-ideal character of the simulation model $f(p_1, \ldots, p_n)$, we may get inconsistency. For example, the program $f(p_1, \ldots, p_n)$ may use Monte-Carlo simulation which produces correct results in a vast majority of the cases, but which sometimes may lead to completely wrong results. In this case, a natural solution is to delete the erroneous intervals, and consider only a consonant sub-collection. To determine which intervals to delete we can use the above algorithm.

The resulting consonant collection has $m < n$ intervals in it. Similarly to the above case, we can assume that all these consonant intervals are equally probable, with probability $1/m > 1/n$. (In effect, what we are doing here is *re-distributing* the probability of the deleted intervals to the remaining ones.)

4.11 Practical Applications

Practical applications of these approaches to different structural design problems are described in [2, 8, 9].

Acknowledgments

This work was supported in part by NASA under cooperative agreement NCC5-209 and NASA/MURED ACE Center, by NSF grants No. DUE-9750858 and CDA-9522207, by the United Space Alliance, grant No. NAS 9-20000 (PWO C0C67713A6), by the Future Aerospace Science and Technology Program (FAST) Center for Structural Integrity of Aerospace Systems, effort sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-95-1-0518, and by the National Security Agency under Grant No. MDA904-98-1-0561.

References

[1] Bojadziev, G., and Bojadziev, M., *Fuzzy sets, fuzzy logic, applications*, World Scientific, Singapore, 1995.

[2] Cooper, A., and Ross, T., *An investigation of new mathematical structures for safety analysis*, Sandia National Lab, Technical Report SAND97-2695, 1997.

[3] Cormen, Th. H., Leiserson, C. E. and Rivest, R. L., *Introduction to algorithms*, MIT Press, Cambridge, MA, and Mc-Graw Hill Co., N.Y., 1994.

[4] Erickson, G. J., Rychert, J. T., and Smith, C. R. (eds.), *Maximum Entropy and Bayesian Methods*, Kluwer, Dordrecht, 1998.

[5] Klir, G., and Yuan, B., *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice Hall, Upper Saddle River, NJ, 1995.

[6] Nguyen, H. T., and Kreinovich, V., "Nested Intervals and Sets: Concepts, Relations to Fuzzy Sets, and Applications", In: R. B. Kearfott et al (eds.), *Applications of Interval Computations*, Kluwer, Dordrecht, 1996, 245–290.

[7] Nguyen, H. T., and Walker, E. A., *First Course in Fuzzy Logic*, CRC Press, Boca Raton, FL, 1999.

[8] Ross, T. J., "Possibility intervals in the characterization of dynamic model uncertainty", *Proceedings of the 19th International Conference of the North American Fuzzy Information Society NAFIPS'2000*, Atlanta, Georgia, July 13–15, 2000, pp. 353–357.

[9] Ross, T. J., and Donald, S., "A new method for the development of empirical probability distributions", *Proceedings of the 2000 World Automation Congress*, Maui, Hawaii, June 11–16, 2000 (to appear).

[10] Wadsworth, H. M. Jr., ed., *Handbook of Statistical Methods for Engineers and Scientists*, McGraw-Hill Publishing Co., New York, 1990.

[11] Yager, R. R. et al. (Eds.), *Advances in the Dempster-Shafer Theory of Evidence*, Wiley, N.Y., 1994.