UNCERTAINTY REASONING
IN EXPERT SYSTEMS

Final Technical Report

Type of report:
Brief Summary of the Project

Principal Investigator:
Vladik Kreinovich

Period Covered:
9/1/1990 – 3/31/1993

Grantee Institution:
The University of Texas at El Paso
500 University Avenue
El Paso, TX 79968

Grant Number:
NASA Grant No. 9-482
Intelligent control is a very successful way to transform the expert’s knowledge of the type “if the velocity is big and the distance from the object is small, hit the brakes and decelerate as fast as possible” into an actual control. To apply this transformation, one must choose an appropriate methods for reasoning with uncertainty, i.e., one must:

1) choose the representation for words like “small”, “big”;
2) choose operations corresponding to “and” and “or”;
3) choose a method that transforms the resulting uncertain control recommendations into a precise control strategy.

The wrong choice can drastically affect the quality of the resulting control, so the problem of choosing the right procedure is very important. From a mathematical viewpoint these choice problems correspond to non-linear optimization and are therefore extremely difficult.

In this project, we develop a new mathematical formalism (based on group theory) that allows us to solve the problem of optimal choice and thus:

1) explain why the existing choices are really the best (in some situations);
2) explain a rather mysterious fact that fuzzy control (i.e., control based on the experts’ knowledge) is often better than the control by these same experts;
3) give choice recommendations for the cases when traditional choices do not work.

Perspectives of space applications will be also discussed.

**Keywords.** Uncertainty, fuzzy control, optimization, stability, smoothness, space applications.

**CONTENTS**

1. Formulation of the problem ................................. 2
2. Main objectives of this research project ...................... 5
3. Main results ......................................................... 6
4. Additional results ................................................. 9
5. List of papers published under the grant ..................... 12
Appendix 1: List of graduate theses and projects defended under the grant ...... 17
Appendix 2: Copies of the papers published under the grant ....... 1–1, ..., 2–1, ...
1. FORMULATION OF THE PROBLEM

Uncertainty reasoning is vitally important for space exploration. The high cost and importance of space missions lead to the necessity for automated systems that help human operators in unexpected situations.

In addition, the distance between the Earth mission control centers and the spacecraft demands that some intelligent software also be present on board the mission in order to make quick decisions in emergency situations.

One of the main problems in solving such control problems (and therefore in creating the corresponding software) is that we have to devise solutions on the basis of an uncertain knowledge of the situation. Thus it is necessary to have methods which allow representation of this uncertain knowledge and which allow automated conclusions based upon such uncertain knowledge.

Two main types of uncertain reasoning. Several techniques have been developed for expressing uncertainty in intelligent systems (see, e.g., a survey [88]). In the majority of these techniques, uncertainty of an expert statement $A$ is represented by a number from 0 to 1: 1 means that we are absolutely sure that this statement is true; 0 means that we are absolutely sure that $A$ is false, and numbers from 0 to 1 represent intermediate degrees of belief. These methods can be divided into two big groups:

- methods in which these numbers are interpreted and processed as probabilities (e.g., probabilistic logic, Dempster-Shafer formalism), and
- methods that use non-probabilistic processing techniques (certainty values, fuzzy techniques).

For both types of methods, uncertainty of expert statements leads to uncertainty in the final decisions.

Probabilistic methods. If we use probabilistic methods to process initial uncertainties, then (in the majority of cases) the formulas for the resulting uncertainties are known from probability theory. The main problem with these methods is that often the existing algorithms are too slow, and therefore, faster algorithms are required if we want to apply these methods to real-time control problems.

Non-probabilistic techniques. For non-probabilistic techniques, there is another important problem: since we are not using probabilistic formulas, what formulas to use? In particular, we must make the following choices:

1) First, we must choose a method to represent the initial expert statements. For each term like “small”, “medium”, etc, that an expert uses to express his knowledge, and for every possible value $x$ of the corresponding variable, we must express his degree of belief that this $x$ is small by a number $\mu(x)$ from an interval $[0,1]$.

2) Second, we must choose a method to combine the resulting degrees of belief that corresponds to “and” and “or”, e.g., what is our degree of belief that $x$ is “small” and velocity $\dot{x}$ is “medium”?
Using these two stages, we are able, for each possible value \( u \) of control, to compute the resulting degree of belief \( \mu_C(u) \) that \( u \) is appropriate for this control situation.

3) Third, we must transform this uncertain knowledge about \( u \) into a single value \( \bar{u} \); this procedure is called defuzzification.

Methods for using uncertain knowledge in control are highly successful. Practically all of the proposed methods of intelligent control have been experimentally tested and proved to be appropriate for some real-life situations (see, e.g., surveys [S85], [L90], [B91], [K92]). Experiments performed at Johnson Space Center on the Shuttle and rover simulators, showed that these methods really lead to high quality control of space missions and planet rovers [L88], [L89], [LJ90].

The choice of uncertainty representation is vitally important for an expert system. This importance is demonstrated by the history of the first efficient expert system - MYCIN [S76]. The operations for combining degrees of belief that are implemented in this system are very complicated. The reason is that while creating MYCIN the authors tried to choose an uncertainty representation that would make the percentage of correct diagnoses higher. They experimented with several formulas, spent a lot of time and money and came to the ultimate formulas that are now implemented. This stage of creating the expert system turned out to be time-consuming but necessary, because all other choices essentially decreased the efficiency of this system - and sometimes even made it useless.

Unfortunately the result of this experiment turned out to be essentially dependent on the concrete domain: the same formulas that are extremely efficient in MYCIN fail in application to other domains. So the problem of choosing the appropriate uncertainty representation technique is really vitally important, because the wrong choice can lead to complete failure. Further, the proper choice can lead to an essential increase in the efficiency of the system - an increase that can otherwise be obtained only by additional expenditures on hardware.

**Trial-and-error choice is impossible for a space mission.** Usually the choice of an appropriate technique is made on a trial-and-error basis, but this is impossible for a billion-dollar project. So, we need theoretical methods for this choice.

References


2. MAIN OBJECTIVES OF THIS RESEARCH PROJECT

The eventual goal of this research is to develop methods for choosing the appropriate representation of uncertainty (either from one of the already existing formalisms or by developing a new one).

In order to achieve this goal, it is necessary to accomplish the following:

• To give a survey of the existing uncertainty reasoning techniques.

• To describe characteristics of different uncertainty reasoning techniques that are maximally relevant to our engineering problems.

• To describe algorithms that estimate the values of the chosen characteristics for different uncertainty reasoning techniques.

• To formulate the problem of finding the best technique as a mathematical problem, and

• To solve this optimization problem, i.e., to find uncertainty reasoning techniques that are optimal with respect to different optimality criteria.
3. MAIN RESULTS

The main results of this research project are published in the conference proceedings paper [26], and in other papers published under this grant. With R. Lea from Johnson Space Center, we are currently working on a book that would incorporate all these results. These results include the following:

A survey of different uncertainty reasoning techniques is given in [26]. In this survey, we not only give the list of all possible techniques, but give a theoretical explanation of why exactly these techniques turned out to be workable.

Characteristics of uncertainty reasoning techniques. For every technique, we must:

1) first, elicit the knowledge from the experts,
2) then, process this knowledge using the corresponding techniques,
3) and finally, apply the resulting control to a spacecraft.

For each stage, we can choose a natural criterion that makes this particular stage most successful:

1) The time and effort needed to solicit the knowledge from an expert, can be described by the average number of binary ("yes-no") questions that we need to ask an expert. This number, in its turn, is related to the accuracy with which we need to know the experts' degrees of belief: if the resulting control is sensitive to this accuracy, then we must determine the degree of belief with better precision, and thus, ask more questions.

2) The time spent on the processing stage is computation time of an algorithm.

3) The quality of the resulting control can be characterized by the criteria that are traditionally used in control theory: stability and smoothness.

So, we have five main criteria for choosing uncertainty reasoning techniques:

- entropy, i.e., the average number of binary questions required to complete the knowledge;
- robustness, i.e., sensitivity of the resulting control relative to the inaccuracies in the initial degrees of belief;
- computational complexity, i.e., the time required for the programs to run;
- stability of the resulting control strategies;
- smoothness of the resulting trajectories.

With the exception of computational complexity (that is a well-defined notion), for all other criteria we could not use the existing criteria, and therefore, had to provide appropriate mathematical definitions. For example, in traditional control theory, stability is usually understood as the following condition: after a small fluctuation, when time \( t \to \infty \), the trajectory eventually returns to its original position. However, if this time \( t \) exceeds the time of the space mission, then this theoretical "stability" is of no use. So,
instead of using the existing theoretical criteria, we give a new definition that formalized
the engineering practice rather than the existing theory.

In particular, as a source of the notion of stability, we considered spacecraft orientation
problems, and as a source of smoothness, spacecraft docking problems (see [26] for details).

For entropy, these new definitions are given in [2], [17], [33]; for robustness, in [31],
[32], [TR3], [A14]; for stability and smoothness, in [21], [26].

In real-life situations, we must use a combined criterion to find a reasonable trade-off
between all five main criteria.

**Algorithms that estimate the values of these characteristics:**

- For entropy, such algorithms are presented in [2], [6], [17], [33]. In particular, [2] and
  [17] cover the Dempster-Shafer and probabilistic approaches, and [6] covers the special
case of non-transitive preferences.

- For robustness, methods are presented in [31] and [32].

- **Computational complexity** is not a very serious problem for non-probabilistic methods,
because we can always choose a technique that is reasonably efficient. For probabilistic
methods, however, the situation is radically different: here, the formulas are given, and
we cannot change them at will. Traditional algorithms of Dempster-Shafer approach
require too long computations, and this is one of the main reasons why these methods
are not universally used. In [1], [11], [A10], we describe an alternative computational
algorithm that enables us to use polynomial-time (i.e., feasible) algorithms.

- For stability and smoothness, methods are given in [21] and [26].

**Formulation of the problem of finding the best technique as a mathematical
problem.** As we have already mentioned, in real-life situations, it is reasonable to use
not only the five main criteria, but their combinations as well. What exactly combination
to use depends on a specific problem. To describe a general case, we developed a general
optimization formalism [19], [25], [26] (see also [20], [22], [23], [24]).

This formalism is based on the so-called group-theoretic (symmetry) approach that
has been so successful in modern theoretical physics.

The main idea of applying this approach to non-probabilistic uncertainty is as follows.
In probabilistic case, the value $t(A)$ that is assigned as a truth value to a statement $A$ has
a very precise meaning, e.g., it describes the ratio of cases in which an expert considers
$A$ to be true. In non-probabilistic case, an expert describes his uncertainty in terms of
words of natural languages (“probably”, “maybe”, etc), and how to represent these words
by numbers is not really that important.

Therefore, if we use a reasonable criterion for choosing a technique, it is natural to
expect that the relative quality of different techniques (with respect to this criterion) should
not depend on what exactly mapping from words to numbers we use.
In other words, we have here a family of natural transformations (that transform uncertainty values obtained by using one mapping into values obtained by another mapping), and the ordering between the techniques that corresponds to optimality criteria must be invariant w.r.t. these transformations.

**Optimal choice of techniques.** Group-theoretic approach not only allows us to formulate the family of reasonable criteria, but also to find the techniques that are optimal with respect to these criteria.

As a criterion, we can take many different combinations of main criteria. Since there are infinitely many possible combinations, it is impossible to describe a technique that is optimal for each of these combinations.

So, in this project, we do the following:

- for the *main* criteria, we solve the optimization problem precisely, and find techniques that are optimal with respect to these criteria;

- for *combined* criteria, we describe a family of techniques that are optimal under different combination criteria; then, when a criteria is given, to find a technique that is optimal with respect to this criteria. it is sufficient to test only techniques from this family.

The description of all techniques that can be optimal under reasonable optimality criteria is given in [26].

This family includes all the techniques that have been empirically shown to be good, and also other techniques that are worth trying.

For *main* criteria, the optimal techniques are described in the following papers:

- for *entropy*, in [33]; in particular, we get $\min(a, b)$ for "and", and $\min(a + b, 1)$ for "or";

- for *robustness*, in [31] and [32]; in particular, we get $\min(a, b)$ for "and", $\max(a, b)$ for "or", piece-wise linear membership functions $\mu(x)$, and standard (center-of-mass) defuzzification;

- for *computational complexity*, min and max are evidently the simplest;

- for *stability*, in [26]; in particular, we get $\min(a, b)$ for "and", and $\min(a + b, 1)$ for "or";

- for *smoothness*, in [26]; in particular, we get $ab$ for "and", and $\max(a, b)$ for "or".
4. ADDITIONAL RESULTS

In this research, we also applied group-theoretic and similar techniques to solve additional related problems.

Is expert knowledge really necessary? Many researchers, especially in traditional control community, still doubt that expert knowledge is necessary. Why not make more experiments and determine the properties of the system?

This is rather a fundamental and theoretical question than a practical one. However, we thought that it would be nice to have an answer to this challenge.

To provide such an answer, we analyzed the most general physical systems.

- One argument in favor of expert knowledge is that for some physical systems, there is simply no way to find their description based on the experimental data only. In particular, such an argument was provided for background microwave radiation: supposedly, there is no way to provably confirm whether it is of cosmological origin or not. We analyzed this example in [9], and proved that this in principle, if we have sufficiently accurate experimental data, it will be possible to distinguish between the cosmological and other models of a 3K radiation. (Another example of cognizability of the physical world is given in [10].)

- However, from the viewpoint of computation time that is necessary to make predictions based on the experimental data, we showed that in the general case, this computation time grows exponentially with the size of data, and therefore, this problem is not feasible [8], [28]. This means that to be able to design feasible algorithms we do need expert knowledge in addition to experimental data.

- Another case when expert knowledge is extremely helpful is the case of the so-called inverse problems. This is a generic mathematical term for the problems in which we reconstruct the parameters of the system from the noisy measurement data. Usually, such problems are ill-posed, i.e., small errors in the measured data can lead to large errors in the parameters. In [27] and [16], we prove that with the expert knowledge added, such problems not only become well-posed, but also that we can apply reasonably fast algorithms to solve them. Unlike the above two fundamental theoretical results, this is also a practical result.

If several different techniques are already used to represent the expert knowledge, what is the best way to combine them? Our main result was aimed at the case when we start “from scratch”, i.e., when we first choose the uncertainty representation technique, and then apply this technique to elicit knowledge from the experts and process it. In many cases, however, when we start the problem, we already have some expert knowledge, and this knowledge is already represented by using non-optimal technique. So, in order to apply the better technique, we must first translate this knowledge from one representation into another.
For non-probabilistic uncertainty reasoning techniques, in [20], we applied the group-theoretic approach to describe the best "interface" translation algorithms. For translation between probabilistic and non-probabilistic methods, a translation is proposed in [30].

**How to combine intelligent control with more traditional control techniques.** Traditional control is best developed for linear systems. So, if we have a non-linear system, it is reasonable to apply some non-linear re-scaling of its variables so that after this re-scaling the system will be either linear, or closer to being linear.

Since this re-scaling works for traditional control, it sounds reasonable to apply it also to the case of intelligent control, i.e., control based on the expert knowledge. This idea was used in [34], [A16], and shown to be reasonably efficient. Moreover, we apply group-theoretic methodology to find the optimal re-scaling.

**How to make expert systems smarter?** In the majority of applications of an expert system to control, we just translate the expert's knowledge into an actual control strategy. The resulting control is sometimes worse than the control by a human operator, because an operator can not only apply his rules, but he can also combine them into more complicated ones (i.e., in other words, he can make logical conclusions).

In [18], [A3], [A11], we show how this can done automatically in a general case. The idea developed in these papers comes from the analysis of chemical systems [A7].

For important problems of pattern recognition and cluster analysis, such algorithms are presented in [TR1] and [TR2].

**How to make an expert system learn?** In the above formulation, we analyzed the problem of how to translate the expert knowledge into the actual control strategy. The resulting control strategy is not perfect: first, the expert was not perfect; second, the translation might not grasp some nuances of his knowledge. So, it is desirable to make the resulting automated system learn. How?

Several techniques are known that make intelligent systems learn, among them analytical techniques (i.e., crudely speaking, numerical computations methods), neural networks, genetic algorithms, etc. All these techniques have been successfully applied to tune intelligent control systems.

But here, we encounter the same problem of choice: there are several different neural network techniques, and in some cases, some of them work fine, and some fail to improve the quality of the system. How to choose?

To solve this problem, we applied a similar group-theoretic methodology, and arrived at the following results:

- for the simplest case when we are changing the degree of belief of a single statement, the optimal technique is presented in [23];
- for neural networks, the optimal techniques are presented in [22]; namely, in [22], we describe the optimal choice of a non-linear basic element (neuron);
• for *genetic algorithms*, the optimal techniques are presented in [24]; namely, we describe the optimal choice of re-scaling.

Two additional results about learning are presented in [29], [7], [A8], and [A13]:

• in [29], we describe *how to combine* different learning techniques (namely, neural and analytical);

• in [7] (see also [A8]), we show that neural networks are (in principle) capable to learn anything (if we use an appropriate learning algorithm). In [7], we prove that such a learning procedure is possible, but provide no example. Such an example is given in [A13] (*warning*: this algorithm is not practically efficient; the main reason for providing it was to prove that in principle, such an algorithm is possible).

**Taking into consideration the uncertainty of the expert knowledge and of the measurement results.** As a result of applying uncertainty reasoning techniques, we design a system that transforms the measurement results into the actual control. The resulting values of control are not precise for two reasons:

• first, experts are not absolutely confident in the rules that they use;

• measurement results are not absolutely precise, because every real measuring device is not perfect.

It is therefore important to take *both* uncertainties into consideration when designing an intelligent control system. This is partially done.

In particular, when we analyzed *sensitivity* [31], [A14], we actually considered sensitivity with respect to both types of uncertainty.

For other criteria, we have just started such analysis (see [5], [12–15], [A1–A2], [A4], [A9], [A12], [A17]).

**By-product of this research: group-theoretical approach leads to new algorithms and results.** The same ideas of optimization under uncertainty have also been successfully applied to other problems:

• In [3], we describe an algorithm for *computer graphics* that is better than the existing ones. This algorithm solves the problem that is very important for space applications, with its 3D areas: the problem of rotating an image around an arbitrary axis.

• In [4], we describe an algorithm for image processing from radar measurements. Here, group-theoretic approach is used to justify the choice of an entropy-like function that we optimize to get the best quality of a reconstructed image.

• Other results (not yet published) are contained in the theses defend under this grant.
5. LIST OF PAPERS, PUBLISHED UNDER THE GRANT
(* indicates a student co-author)


**Technical Reports**


**Published Abstracts**


APPENDIX 1. LIST OF GRADUATE THESSES AND PROJECTS DEFENDED UNDER THE GRANT


6. Liz Kamoroff. “How to extract knowledge from an expert so that his effort is minimal”, Master Project, Computer Science Department, University of Texas at El Paso, October 1992.


