Computational Decision Analysis

Mats Danielson
DECIDE Research Group
Dept. of Computer and Systems Sciences
Royal Institute of Technology
Electrum 230
SE-164 40 KISTA
SWEDEN
Phone: +46 8 16 1679
Fax: +46 8 703 9025
email: mad@dsv.su.se
URL: http://www.dsv.su.se/~mad

Ph.D. Thesis
Dept. of Computer and Systems Sciences
Royal Institute of Technology (KTH)
Stockholm, Sweden

ISBN 91-7153-613-2

Printed by Akademitryck, Edsbruk, 1997

To *Elin* and *Martin*

Acknowledgements

First of all, I would like to thank the three members of my thesis committee, who have all been valuable mentors. My formal supervisor, Prof. Carl Gustaf Jansson, has over the years shown great patience and encouragement during my sometimes seemingly homeless graduate studies, as I drifted between subjects. He finally guided me into a safe haven. My thesis supervisor, Docent Per-Erik Malmnäs, Dept. of Philosophy, Stockholm University, made this thesis possible by formulating the original research problem which this work took as its starting point. My thesis co-supervisor, Dr. Love Ekenberg, IIASA, Vienna, has always been available for discussions and has been of great help with his deep insights into both decision theory and life as a graduate student, having wandered the same road a couple of years before me. Sharing an office with him speeded up this work considerably. Dr. Magnus Boman did a great deal of proofreading when the author became completely blind to his writing mistakes. Johan Walter, M.Sc. student, also read parts of the manuscript and co-developed a graphical interface to the DELTA Decision Tool. I would also like to thank Eva Jansson and the other administrative staff at DSV for keeping the project afloat.

Stockholm, February 1997

Contents

* Chapters missing in the PDF reprint since the files could not be found.
The short unnumbered part introductions could not be found either.

If we have been accustomed to deplore the spectacle […] of a workman occupied during his whole life in nothing else but the making of knife handles or pins' heads, we may find something quite as lamentable in the intellectual class, in the exclusive employment of the human brain in resolving some equations, or in classifying insects. […] It occasions a miserable indifference about the general course of human affairs, as long as there are equations to solve and pins to manufacture.

Auguste Comte

Preface

Methods for decision-making are of prime concern to any enterprise, even if the decision processes are not always explicitly or even consciously formulated. All kinds of organisations must continuously make decisions of the most varied nature in order to survive and attain their objectives. A large part of the time spent in any organisation, not least at management levels, is spent gathering, processing, and compiling information for the purpose of making decisions supported by that information. Decision-making has many aspects and this thesis focuses on one of them – modelling and evaluating possible courses of action given imprecise information.

The idea of using computers to support decision-making has been around for a long time, almost as long as computers have been available in any usable form. Some of the more prominent ideas took the form of research in artificial intelligence (AI) and operational research (OR). From a fairly close relationship at the outset, that part of AI research took the more symbolic path while OR stayed on the numerical side as a branch of applied mathematics. Naturally, as computers became an inseparable part of many modelling attempts, the areas of computer science and information systems have also contributed indirectly to numerous approaches. During many years of research, a number of interdisciplinary sub-fields have emerged as responses to decision needs, each addressing its particular class of problems. The distinction is not sharp and ideas from more than one sub-field as well as from other sources can be found in some approaches.

The area of *decision tools* contains approaches dealing with mechanising the structuring and analysis of decision situations. One of the ideas is to model the situations according to some normative model of rational decision behaviour. Presuming the decision-maker to be rational, the mechanical model can devise suitable courses of action given the supplied information. This approach does not require the tool to possess any degree of specialised expertise in the target area of the decision. Tools can be analytic, in the sense that they handle a smaller number of alternative courses of action and support the evaluation and selection of those alternatives. Synthetic tools, on the other hand, handle a larger number of alternative courses of action, and instead support the design of problems by, for example, generating alternatives.

The two close areas of *decision support systems* and *management information systems* partly deal with collecting large amounts of information, predominantly in professional organisations. The collected information is then subject to statistical and other quantitative analyses in order to extract and compress information to aid management in making better decisions. There is a focus on which information to extract, how to extract it, and in which form to present it.

In later years, there has been a growing interest within the field of artificial intelligence (AI) in the well-founded area of decision theory, which has merged with other uncertainty techniques into the sub-field of uncertain reasoning [SP89]. In the 1980s the area of knowledge-based systems (KBSs) grew strong within AI. The idea with KBSs is to supply the user with a package of specialist knowledge in a particular area that can be consulted when facing problems that require expertise beyond the decision-maker's own level. Using AI techniques for knowledge representation, search, and inference, a KBS is supposed to act as the expert consultant in a decision situation and supply the decision-maker with professional advice.

Another AI area, probabilistic reasoning, deals with automated reasoning in domains represented by probabilistic networks.[1] The reasoning can be divided into an inference part and a decision part. The inference part is the larger one in the sense that more research efforts have gone into it so far. It deals with inferring probabilistic support for conclusions drawn from collected evidence. The decision part, which is most closely connected to the topic of this thesis, consists of value nodes present in the network. The expected values of those nodes should be maximised subject to the constraints of the network in the form of arcs between chance and decision nodes. One of the most popular methods is the influence diagram technique. The diagrams are used for evaluating decision situations arising in inference networks. They combine inference and decision into one formalism, and they offer space-saving techniques for representing elaborate probabilistic graphs. The evaluation can be performed in various ways, classical ones being to convert the diagram to a decision tree [HM84] or to transform the diagram while preserving the expected value [S86].

Other AI techniques employed in various decision systems include machine learning techniques such as inductive decision trees, neural networks and genetic algorithms. Those are all good candidates for automated decision systems since they have strong predictive abilities. As methods for general decision analysis tools, they are less promising since they do not offer enough insight into or exploration of the decision problem. This might change, however, as the cross-fertilisation between different approaches to solving decision problems continues.

Mechanising Decision Analysis

Why does anyone want to have a *mechanical* method for decision analysis? Should not all decisions, at least by humans but possibly also by AI systems, be based on experience, intuition, and sound judgement?

[1] For a general introduction to probabilistic reasoning, see for example [N90, P91].

Many human decision-makers will undoubtedly react negatively to the idea of being replaced by a computer program, and not without reason. Management information systems of previous decades have not fulfilled their promises to any great extent [MM73]. In this thesis, a decision method called DELTA and belonging to the area of analytic decision tools is presented. The purpose of the method is not to replace human decision-makers with machines, nor to replace any AI techniques. On the contrary, the objective is to increase the decision-maker's (or module's) ability to make sound decisions. A framework in which to express the decision problem and a clearly defined process helps in understanding the decision situation. It also provides a good overview of the decision material.

Given a decision situation and some statements of probabilities and values, the method will indicate preferred ways to act. Moreover, it may point out weaknesses in the underlying information. The approach is interdisciplinary and draws on ideas from AI and OR as well as from statistical decision theory. It allows the decision-maker to be as deliberately imprecise as he feels is necessary and provides him with the means for expressing varying degrees of imprecision in the input statements. This leads to a more natural relationship between the decision-maker and the support tool.

In many attempts to find general methods for solving different decision problems, one of the common denominators has been the imprecise nature of the input data. Regardless of the method employed, some kind of sensitivity analysis must be carried out in which the proposed solution is exposed to various what-if tests. Traditionally, these tests are done in a low-dimensional fashion, studying one or at best a few of the input variables at a time. This yields only limited insight into the problem since the full impact of imprecision on the solution cannot be appreciated. A feature of the DELTA method is an automated way of carrying out multi-dimensional analyses of the obtained results.

In addition, it should be noted that the decision-maker is in control of the flow of events. He can choose to say as much or as little as he wishes about each particular piece of information. He can also make his statements in any order since the method is not the controlling part of the interaction process. The method acts instead as an aid, presenting evaluations of alternatives and guiding the decision-maker in his search for good courses of action. This contrasts with many KBSs, which often control the flow of interaction and contain expert information the user is not supposed to possess. A traditional KBS often draws conclusions and makes decisions over which the user is not in full control and may not even fully understand. The decision method proposed in the following chapters encourages an exploratory style of working, which seems to correspond well with the way decision-makers reason without the aid of computer tools. One of the objectives is therefore to provide a tool that supports rather than controls the intended user, and that makes the greatest possible use of his own expertise in the target area. For human decision-makers, the target is often complex. Such decisions are made in the heads of the decision-makers, and the role of a tool is more to present an analysis and offer simulations or sensitivity analysis to aid the understanding of the problem and the decision situation.

Contributions

The seeds for the research problems in this thesis were originally discussed by Per-Erik Malmnäs in his Ph.D. thesis [M81] and further elaborated by him during the 1980s. Most of the research for this thesis has been carried out within the DECIDE Research Group. In the early 1990s, the **μ** decision method was suggested by Per-Erik Malmnäs [M90], refined by Love Ekenberg in his Ph.D. thesis [E94], and implemented as the **μ** decision solver by the author at around the same time [D93]. Since then, another approach has been developed by the author into the comprehensive and fully computational DELTA method presented in this thesis.

The research was carried out at the Department of Computer and Systems Sciences (DSV), KTH, during a period of more than three and a half years from April 1993 to November 1996. The work has been partially supported by NUTEK. It started in an effort to increase the speed of decision evaluation algorithms and ended up becoming a new computational framework for decision analysis [D97b].

Main developments have taken place within the areas of representing knowledge, determining properties of bases (collections of constraints), evaluating consequence sets, and computing the results. The thesis contributes to previous research, within and outside of the group, in at least the following ways.[2] Perhaps the major contribution lies in presenting a framework and a complete, implemented method for decision analysis using imprecise input data. Most other similar efforts either concentrate more on representation than on evaluation or are never fully implemented. Without implementations, only small problems can be handled and studied, and it is not easy to assess such attempts. In a recent survey of tools available from labs worldwide in a closely related field, multiple criteria decision aids, Olson finds only a few tools that handle imprecise statements at all [O96].

The DELTA method includes a number of established ideas and concepts, but many are new or generalised. The following more detailed account tries to highlight some of the advancements. The method has a novel representation using the concept of interval statements, which greatly simplifies the presentation. In representing such statements, the introduction of core estimates in addition to constraints is new and enables a much clearer usage of this kind of representation in allowing the expression of imprecision in several ways. Using only constraints is often not enough – they result in too wide, overlapping evaluation results. Core estimates admit a more powerful representation by allowing both positive and negative statements. Some other representation concepts have been reworked since [D95], such as the orthogonal

[2] All contributions are discussed relative to terminology partly introduced later.

hull. The classification based according to their computational demands is also new, enabling a hierarchy of solver algorithms to be set up. New concepts for bases include their symmetry and skewness, and the symmetric hull as well. This facilitates different views on the input data. The concept of proportion [M90, E94] is replaced by the new concepts of expansion and contraction [D97b], enabling sensitivity analyses of many input statements at a time. Many statements made by decision-makers are given new interpretations (translations).

The evaluation rules of collections of imprecise statements usually concentrate on some notion of admissibility, the classical standpoint being summarised in for example [L59]. The suggestions by some well-known researchers are of this kind [L74, GS82], while others have taken the concept further, for example by extending it with a parameter into t-admissibility [M94a, E94]. This thesis takes a broader view in trying to generalise and integrate many known numeric decision rules into the computable concept of Δ-dominance. Further, a new set of selection rules is made possible by the introduction of the concepts of strong, marked, and weak dominance. Those concepts together with expansions and contractions of the bases enable a family of evaluation principles.

The evaluations would not be of much interest if they were not efficiently computable, and one chapter is devoted to the optimisation of linear and bilinear problems. The algorithms presented include the solution of important classes of the bilinear programming problem by reduction to linear programming problems. Problems of determining properties of collections of decision-maker statements are mapped onto the well-known mathematical theory of linear programming. In order to evaluate the various properties, the Simplex method is employed. In general, Simplex research is focused on solving very large systems of inequalities, and not much research has gone into using it for solving long sequences of smaller problems. This thesis focuses on this more unusual problem setting and evaluates different techniques originally proposed for solving standard linear programming problems.

Implementations of the algorithms for many versions of the DELTA method were made and they were run on several machines. They resulted in the DELTALIB library with a layered architecture, see Figure 3.1 in Chapter 3, where modules can be exchanged for the purpose of testing new algorithms. Implementing algorithms for those modules led to the completion of a solver package subsequently used in real-life applications. Many insights into what to improve were gained from that effort. For the current DELTA solver, an extensive series of experiments was conducted with the aim of finding efficient algorithms for use on a wide range of computers. Some of the measurements were reported in [D95]. The current implementation, comprising some 15,000 lines of source code, has been written in portable ANSI-style C using an object-oriented design method to make a transfer to almost any operating system possible. The Simplex method in particular required many alternative implementations, as it is sensitive to the relative speed of various elementary computer instructions.

Publications

In my licentiate thesis [D95], the research directions were stated but the results were incomplete. Especially the properties of the representation (Chapter 4) and the evaluation rules (Chapter 5) were partial at that time, covering only some cases of the now much more complete DELTA method. The licentiate thesis was written in late 1994 and early 1995, and the work with the completion of a version of the method was carried out during the rest of 1995 and the first three quarters of 1996. The core of the licentiate thesis and some of the 1995 results were submitted to the European Journal of Operational Research. The article is accepted for publication and will appear in 1997 [DE97c].

Implementations of the algorithms of DELTA are collected into DELTALIB. A decision tool (Chapter 3) built on top of the library was presented at the IIASA Workshop on Advances in Methodology and Software for Decision Support Systems in Vienna in September 1996.[3] The presentation is available as a DSV report [D96].

The applications to distributed AI (multi-agent systems) are covered in a series of articles that mix older ideas from Ekenberg's Ph.D. thesis with the newer DELTA method. There are two journal articles, one in International Journal of Cooperative Information Systems in 1996 [EDB96], and one in Decision Support Systems International Journal, accepted and to appear in 1997 [EDB97]. Appendix A is a revised version of both papers, updated to discuss the DELTA application in more detail.

The applications to risk management have been published in an earlier article on computer security [ED95] and a more recent journal article on general risk analysis in Scandinavian Insurance Quarterly 1996 [DEE96]. The latter describes a comprehensive risk analysis method with DELTA as the evaluation tool. Appendix B is a translation and development of that article.

Finally, in the section on further research in the Conclusion, one direction is extending DELTA into the multi-criteria area. The first attempt was described at one of the largest multi-criteria conferences in January 1997 [DE97a]. Those results are not included in this thesis. Here a decision model and decision analysis refer to decisions under a single criterion. The author was also the grant receiver and project leader of a NUTEK project using multi-criteria decision analysis in evaluating the alternatives in a procurement of railway equipment [DE97b]. Due to the nature of the project, the detailed results is classified information. They are not publicly available and thus not included in this thesis. However, the project report describes the method used and outlines the results.

[3] IIASA is the International Institute for Applied Systems Analysis in Vienna.

Applications

The DELTA method and its predecessors have been used in a number of different real-world applications. The word *application* itself can mean at least two things in this context. In one sense, it can be taken to mean applying the evaluation framework to different subject areas. Apart from traditional decision analysis, this has been done successfully in a number of cases, notably in the areas of multi-agent systems [EBD95, EDB96a, EDB96b, EDB97], risk management [ED95, DEE96], and lately multi-criteria decision making [DE97a, DE97b]. In another sense, it can mean applying the proposed decision method to situations where decisions are to be made. This has also been done in a number of cases including choosing computer software packages,[4] selecting national policies for health care,[5] assessing risks in vehicle electronics,[6] and procuring railway equipment (5 billion SEK, mentioned above).[7]

Thesis Structure

The thesis is divided into three main parts: Introduction, The DELTA Method, and Conclusion and Applications. The first part makes up an informal introduction to decision analysis in general and to the DELTA method. Chapter 1 discusses some common decision models and research methods, so as to have something with which to compare the proposed approach. Chapter 2 is a brief introduction to a work process involving the DELTA method, in order to point at one plausible use for the method. Chapter 3 presents DDT – the DELTA Decision Tool – intended for aiding human decision-makers in understanding their decision problems.

---

[4] Trygg-Hansa (Swedish insurance company) [D93, E94].

[5] Socialstyrelsen (National Board of Health and Welfare), unpublished.

[6] Volvo Personvagnar (Swedish car manufacturer), NUTEK project [DE97b].

[7] Banverket (Swedish National Rail Administration), NUTEK project [DE97b].

The second part presents the method in detail and starts in Chapter 4 with the structure of a decision problem and the knowledge representation for the decision statements. The chapter also presents general properties of bases. Following that, it discusses properties particular to probability bases and then to value bases. Finally, it suggests translations of imprecise statements into constraints. In Chapter 5, evaluation rules are investigated. First, evaluation rules in general are discussed, starting with the expected value rule and then continuing with some alternative rules. The unifying concept of Δ-dominance is suggested as the evaluation principle for the DELTA method. Then follows a section on techniques specific to collections of imprecise statements. Chapter 6 deals with optimisation algorithms for the method. It starts with linear programming for determining properties of bases and continues with bilinear programming necessary to calculate the results of the evaluation rules of the preceding chapter. The last section of the chapter describes Simplex as it applies to the DELTA solver. The proofs given in Part II are intended to convey the meaning of and aid in understanding the DELTA method.

The third and last part starts with a summary and some pointers to future research. Next, examples of applications of the method to other areas are included in two appendices. Appendix A brings up the topic of multi-agent systems and the applicability of DELTA to that area. The other application, Appendix B, concerns the area of risk analysis as the concept is understood within insurance and security. Finally, the thesis ends with references, lists of definitions, examples, figures and tables in the thesis, and an index.[8]

[8] Minor language adjustments and corrections from the thesis errata sheet and the thesis defence have been incorporated into the reprint.

And then one day you find
Ten years have got behind you
No one told you when to run
You missed the starting gun

You run and you run
To catch up with the sun
But it's sinking
Racing around
To come up behind you again

The sun is the same
In a relative way
But you're older
Shorter of breath
And one day closer to death

– R. Waters

2

Working with DELTA

This chapter is an introduction to a proposed human decision work process in which the DELTA method plays a central role. It is intended to serve only as an informal overview, introducing ideas and terminology enlarged on in Part II. The purpose is not to describe the mathematical or computational machinery necessary, but rather to give an intuitive overview of how the method works and of its relevance to organisational decision-making. Another objective is to demonstrate that the suggested method is realistic to work with.

A feature of the method is that the decision-maker has to make his problem statements more visible than he would otherwise. This brings about a number of advantages. First, he must make the underlying information clear, and second, the statements can be the subject of discussions with (and criticism from) other participants in the decision process. Third, it can also be seen more clearly which information is required in order to "solve" the problem and within which areas some more information must be gathered before a well-founded decision can be made. Fourth, arguments for (and against) a specific selection can be derived from the analysis material. Fifth, the decision can be better documented, and the underlying information, as well as the reasoning leading up to a decision, can be traced afterwards. The decision can even be changed in a controlled way, should new information become available at a later stage.

Professional decision-makers in corporations as well as in public organisations today often use rather simple decision models to aid decisions. In many cases, decisions are made without employing any model at all. The decision might be based on rules of thumb or on intuition, or even be a repetition of a similar decision made earlier. Sometimes, decisions are made after listing the alternatives and discussing their consequences in an unstructured manner. These alternative–consequence lists may state the advantages and disadvantages of each course of action. When the special case of one action having all advantages and another all disadvantages does not prevail, it is often necessary to make a complicated comparison between the consequences of all alternatives. Other examples of well-known traditional decision aids include decision matrices and decision trees as discussed in Chapter 1. Many of them have the common disadvantage that they either do not handle probabilities at all, or else they require the decision-maker to make probability statements with precise numeric values, however unsure he is of his estimates.

Suppose a decision-maker wants to evaluate a specific decision situation. In order to solve the problem in a reasonable way, given available resources, a decision process such as the following could be employed, not necessarily in the exact order given.

- Clarify the problem, divide it into sub-problems if necessary
- Decide which information is a prerequisite for the decision
- Collect and compile the information
- Define possible courses of action
- For each alternative:
  - Identify possible consequences
  - For each consequence:
    - If possible estimate how probable it is
    - If possible estimate the value of it occurring
- Disregard obviously bad courses of action
- Based on the above, evaluate the remaining alternatives
- Carry out a sensitivity analysis
- Choose a "reasonable" alternative

The model described in the following should be seen in the context of such a decision process. The process is intuitively appealing, and numerous decision-makers unconsciously use a similar approach.

The Work Cycle

The decision process is carried out in a number of steps presented here in work-cycle form. A *work cycle* consists of six phases (Figure 2.1). The first step of the first cycle is special since there is much information to collect. The initial information is gathered from different sources. Then it is formulated in statements as indicated later in the chapter and entered into the DELTA Decision Tool (DDT, see Chapter 3).[1] Following that, an iterative process commences where step by step the decision-maker gains further insights and sometimes a conclusion. During this process, the decision-maker receives help in realising which information is missing, is too vague, or is too precise. He might also change the problem structure by adding or removing consequences or even entire alternatives as more decision information becomes available.

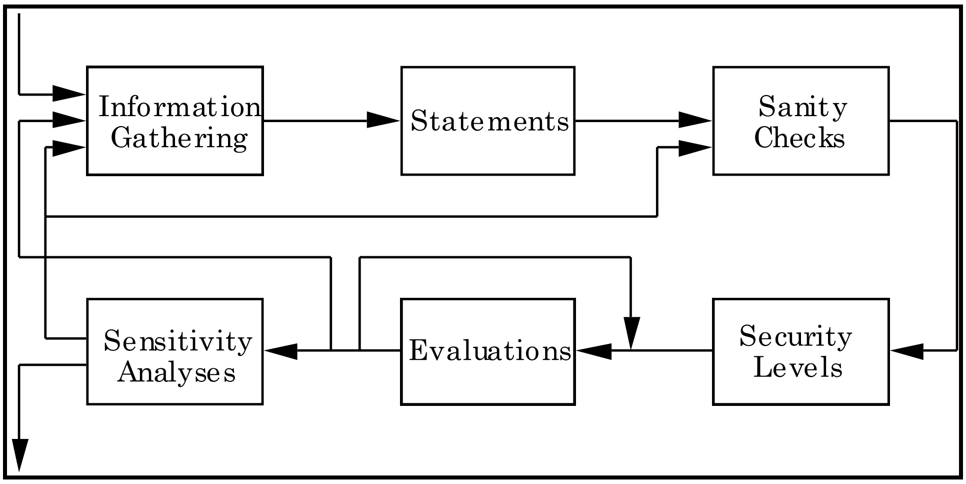

Figure 2.1 The DELTA work cycle

[1] The current version of DDT accepts numeric input by rulers, while future versions will accept linguistic input as well.

Information Gathering

In some cases, the first information collection phase can be a very long and tedious step. In larger investigations, it might take many man-years and result in documentation covering several meters of shelf space. In other cases, it might only require a few half-day discussions with experts and affected workers. It is impossible to describe any typical case because the situations are too diverse.

After the data collection phase, a filtering task commences where the decision-maker structures and orders the information. He tries to compile a smaller number of reasonable courses of action and identify the consequences belonging to each alternative. There is no requirement for the alternatives to have the same number of consequences. However, within any given alternative, it is required that the consequences are exclusive and exhaustive, i.e. whatever the result, it should be covered by the description of exactly one consequence. This is unproblematic, since a residual consequence can be added to take care of unspecified events.

Statements

Once the information is structured, it is entered into DDT in the form of statements such as *the probability of consequence C occurring is less than 40%*. For each new statement entered, the consistency of the information is checked.

The decision-maker's probability statements are represented by interval constraints and core intervals as further described in Chapter 4 on representation. Intervals are a natural form in which to express such imprecise statements. It is not required that the consequence sets are fixed from the outset. A new consequence may be added at a later stage, thus facilitating an incremental style of working. The collection of probability statements in a decision situation is called the *probability base*. Some elementary statements considered are the following.

- *The event H_1 is probable*
- *The event H_1 is possible*
- *The event H_1 is improbable*
- *The probability for event H_1 is a*
- *The probability for event H_1 is larger than a*
- *The probability for event H_1 is between a and b*
- *The event H_1 is as probable as H_2*
- *The event H_1 is more probable than H_2*
- *The event H_1 is much more probable than H_2*

A probability base is said to be *consistent* if it can be assigned at least one real number to each variable so that all inequalities are simultaneously satisfied.[2] The idea is that no meaningful operations can take place on a set of statements that have no variable assignments in common, since there is no way to take all the requirements into account. Note that the method deals with classes of functions of which there are infinitely many instantiations, and insists on at least one of them yielding consistent results.

Likewise, the values are expressed as interval statements. The translations of the value statements in a decision situation are called the *value base*. Some elementary statements considered in this thesis are the following.

- *The event H_1 is desirable*
- *The event H_1 is acceptable*
- *The event H_1 is undesirable*
- *The value of event H_1 is a*
- *The value of event H_1 is larger than a*
- *The value of event H_1 is between a and b*
- *The events H_1 and H_2 are as desirable*
- *The event H_1 is more desirable than H_2*
- *The event H_1 is much more desirable than H_2*

[2] For example $p(H_1) = 0.22$ and $p(H_2) = 0.39$.

Consistency is defined in the same way as for a probability base, and is also discussed in Chapter 4. The probability and value bases together with structural information constitute the *decision frame*.

When all statements in the current cycle have been entered, the data entry phase is over. As the insights into the decision problem accumulate during all the following phases, it is possible to add new information and alter or delete information already entered.

Sanity Checks

Thereafter, the work cycle goes into evaluating the alternatives. The first cycle begins by comparing the alternatives as they are entered. As the first evaluation step, the *sanity* of the decision frame is checked. Much information collected, especially in large investigations, runs the risk of being cluttered or misunderstood during the process. If some data in the frame is problematic, the decision-maker could consider leaving it out of the current cycle or recollecting it. Missing data is easily handled for later inclusion. For example, a missing consequence can be added at a later stage. If the set of consequences for some alternative is not exhaustive, a residual consequence can be temporarily added. Missing value constraints can be temporarily substituted with very wide intervals or just left out. Such possibilities have certain advantages as the results emerging at the outset of the evaluation may be viewed with greater confidence than if erroneous data is entered.

Security Levels

Many decisions are one-off decisions or are important enough not to allow a too undesirable outcome regardless of its having a very low probability. The common aggregate decision rules will not rule out an alternative with such a consequence provided it has a very low probability. If the probability for a very undesirable consequence is larger than some *security level*, it seems reasonable to require that the alternative should not be considered, regardless of whether the expected value shows it to be a good course of action. If the security level is violated by one or more consequences in an alternative and this persists beyond a predetermined rate of contraction (described below), then the alternative is *unsafe* and should be disregarded. An example of security levelling is an insurance company desiring not to enter into insurance agreements where the profitability is high but there is a very small but not negligible risk for the outcome to be a loss large enough to put the company's existence at stake. The security analysis requires some parameters to be set. This can often be done at an organisational level, and it will then have the effect of creating a policy within the organisation. Security levels is an important supplement to the expected value. It is more formally introduced in Chapter 5 and further discussed and exemplified in Appendix A.

Evaluations

After having taken security levels into account, which value does a particular decision have? In cases where the outcomes can be assigned monetary values, it seems natural that the value of the decision should be some kind of aggregation of the values of the individual consequences. One suggestion is to assign different weights to the consequences so that more probable ones are more influential than less probable ones. This line of reasoning leads to the expected monetary value (EMV), which is essentially the same construct as the general expected value discussed below. EMV shows the monetary result that would be obtained on average, should the decision situation reoccur a large number of times. Since not all decisions reoccur that often, some not at all, EMV should be interpreted as the average tendency prevailing in every decision situation.

There are a number of possible evaluation rules within DELTA, some of which are described in Chapter 5. Often, the final comparing rule of an evaluation in the DELTA method as well as in many other methods is the expected value (EV), sometimes instantiated as the expected utility or the expected monetary value. Since neither probabilities nor values are fixed numbers, the evaluation of the expected value yields quadratic (bilinear) objective functions of the form

$$EV(A_i) = p_{i1}v_{i1} + \dots + p_{in}v_{in}$$

where the p_{ik}'s and v_{ik}'s are variables. Maximisations of such expressions are computationally demanding problems to solve in the general case, using techniques from the area of quadratic programming [L89]. In Chapter 6 there are discussions about and proofs of the existence of computational procedures to reduce the problem to systems with linear objective functions, solvable with ordinary linear programming methods.

When a rule for calculating the EV for decision frames containing interval statements is established, the next question is how to compare the courses of action using this rule. It is not a trivial task, since usually the possible EVs of several alternatives overlap. The most favourable assignments of numbers to variables for each alternative usually render that alternative the preferred one. The first step towards a usable decision rule is to establish some concepts that tell when one alternative is preferable to another. For simplicity, only two alternatives are discussed, but the reasoning can easily be generalised to any number of alternatives.

> Alternative A_1 is *at least as good as* A_2 if $EV(A_1) \geq EV(A_2)$ for all consistent assignments of the probability and value variables.
>
> Alternative A_1 is *better than* A_2 if it is at least as good as A_2 and further $EV(A_1) > EV(A_2)$ for some consistent assignments of the probability and value variables.
>
> Alternative A_1 is *admissible* if no other alternative is better.[3]

If there is only one admissible alternative it is obviously the preferred choice. Usually, there are more than one since apparently good or bad

[3] This conforms to statistical decision theory [L59].

alternatives are normally dealt with on a manual basis long before decision tools are brought into use. All non-admissible alternatives are removed from the considered set and do not take further part in the evaluation. The existence of more than one admissible alternative means that for different consistent assignments of numbers to the probability and value variables, different courses of action are preferable. When this occurs, how is it possible to find out which alternative is to prefer?

Let $\delta_{12} = EV(A_1) - EV(A_2)$ be the differences in expected value between the alternatives. The *strength* of A_1 compared to A_2, given as a number $\max(\delta_{12}) \in [-1,1]$, shows how the most favourable consistent assignments of numbers to the probability and value variables lead to the greatest difference in the expected value between A_1 and A_2. In the same manner, A_2 is compared to A_1. These two strengths need not sum to one or to any other constant – the first might for example be 0.2 and the second 0.4. If there are more than two alternatives, pairwise comparisons are carried out between all of them.

Furthermore, there is a strong element of comparison inherent in a decision procedure. As the results are interesting only in comparison to other alternatives, it is reasonable to consider the differences in strength as well. Therefore, it makes sense to evaluate the *relative strength* of A_1 compared to A_2 in addition to the strengths themselves, since such strength values would be compared to some other strengths anyway in order to rank the alternatives. The relative strength between the two alternatives A_1 and A_2 is calculated using the formula

$$\text{mid}(\delta_{12}) = \frac{\max(\delta_{12}) + \min(\delta_{12})}{2} = \frac{\max(\delta_{12}) - \max(\delta_{21})}{2}$$

which is explained in detail in Chapter 5. The concept of strength is somewhat more complicated than discussed in this chapter. Alternative A_1 is said to strongly dominate alternative A_2 if $\min(\delta_{12}) > 0$, to markedly dominate if $\text{mid}(\delta_{12}) > 0$, and finally to weakly dominate if $\max(\delta_{12}) > 0$.[4] This is also explained in Chapter 5.

Only studying the differences in the expected value for the complete bases often gives too little information about the mutual strengths of the alternatives. Numbers close to any of the boundaries seem to be the least reliable ones when making the original imprecise statements. Hence, it would be advantageous to be able to study the strengths (or dominances) between the alternatives on sub-parts of the bases. If a dominance is evaluated on a sequence of ever smaller sub-bases, a good appreciation of the strength's dependency on boundary values can be obtained. This is denoted *contracting* the bases, and the amount of contraction is indicated as a percentage which can range from 0% to 100%. For a 100% contraction, the bases are contracted into single points, and the evaluation becomes the calculation of the ordinary expected value.[5]

The next chapter presents the DDT tool in some detail, complete with evaluation graphs. The results of the comparisons can be displayed either in a diagram for each pair of alternatives or as a summary for each alternative.

Sensitivity Analyses

After the evaluation, a *sensitivity analysis* is the next step. The analysis tries to show what parts of the given information are most critical for the obtained results and must therefore be given extra careful consideration. This is accomplished by varying a number of statements in desired ways, increasing or decreasing intervals, modifying structural information, etc. It also points to which information is too vague to be

---

[4] To be more precise, the DELTA method uses the concept of Δ-dominance as described in Chapter 5. It may colloquially be interpreted as the relative strength between the alternatives.

[5] The method uses the dual concepts of expansion and contraction as explained in Chapters 4 and 5, but the idea is the same as only contracting the bases. Since the core is not discussed in this chapter, neither is expansion.

of any assistance to the ongoing evaluation. Information identified in this way is subject to reconsideration, thereby triggering a new work cycle.

It is possible to regard the expansion and contraction procedures as automated kinds of sensitivity analysis. In order to maintain consistency, the expansion (contraction) increases (decreases) the bases in predefined ways. The decision-maker might, however, have other ideas of interesting modifications to make to the bases, like decreasing or even increasing selected intervals. He might have structural or problem specific information that leads him to manipulate certain intervals in special ways. A common strategy is decreasing intervals until only one alternative is admissible. This way further insights into the decision problem can be gained. It is simple to allow for this in the DELTA method and the procedures of expansion and contraction apply equally well to bases altered for reasons of sensitivity analysis.

Before a new cycle starts, alternatives found to be undesirable or obviously inferior by other information are removed from the decision process. Likewise, a new alternative can be added, should the information gathered indicate the need for it. Consequences in an alternative can be added or removed as necessary to reflect changes in the model. Often a number of cycles are necessary to produce an interesting and reliable result.

Decision Process Results

After the appropriate number of work cycles has been completed, both the decision problem and its proposed "solution(s)" in the form of preferred courses of action will be fairly well documented. Anyone interested and with access to the information can afterwards check, verify (and criticise) the decision based on the output documentation, which because all consequences are clearly presented shows how all the alternative courses of action have been valued. Also, during the decision process, the analysis is open for comments and can become the basis for further discussions. Another effect is that the decisions are less dependent on which employee handles a particular decision situation since deviations from corporate policy can be detected in the documentation after the process has been completed if not earlier.

This concludes the informal introduction to the DELTA method in a work process. The next chapter presents the DDT tool suitable for interactive use in a work cycle-based process. The chapters that follow in Part II go into considerably more detail in trying to present the representation and the evaluation procedures of the method.

3

The DELTA Decision Tool

This chapter is a demonstration of DDT – the DELTA Decision Tool. DDT is built on top of DELTALIB, a set of library procedures that together implement the DELTA method as described in Part II of the thesis. The chapter is divided into three sections. The first section describes the DDT software and its architecture. The functionality of the software is most accessibly conveyed by an example. Thus, the middle section introduces a decision problem on which the sample session in the last section is built. The chapter is intended to continue the informal overview from Chapter 2. As in that chapter, the purpose is to provide an intuitive overview of how the method works and to demonstrate that the suggested method is realistic to work with.

The DDT Software

The DELTALIB library is the core of DDT [D96]. It consists of several modules collected into a library with a common published programming interface in the form of callable C functions and procedures. The layered architecture of the library is shown in Figure 3.1. The lowest layer, the *solver layer*, consists of different optimising solvers for linear and bilinear programming as described in Chapter 6. There is a solver stack consisting of a number of solvers that solve progressively harder problems of optimisation. Further, there resides other algorithms such as graph algorithms for special purposes.

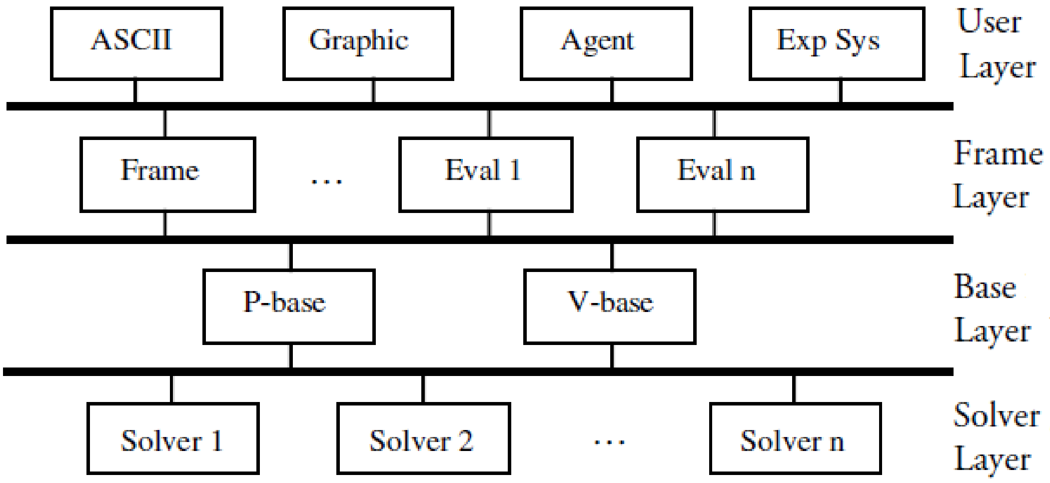

Figure 3.1 The DELTALIB layers

The next layer, the *base layer*, contains functions for the probability and value bases. Among the functions are data structure access, consistency maintenance, and tests for orderings. This layer calls the solver for tasks that involve optimisation, for example calculating the orthogonal hull of the probability base.

The *frame layer* is the library's interface to the callers. It provides a programming API[1] to the library functions and capabilities. It contains a scheduler, consistency and maintenance functions, and integrity checks to protect the rest of the library from erroneous calls. Further, it contains the evaluation modules for the DELTA and GAMMA rule sets (explained in Chapter 5), and for the PSI and OMEGA rule sets not explicitly covered in the thesis. Finally, it contains the procedures for security levels. The layer may be extended with other functions in the future, for example evaluations using other Δ-dominance concepts or numerical rules other than the expected value.

The *user layer* consists of different library users. The library is equally well designed for use by a textual user interface, a graphical user interface, an agent (a robot, for example), or an expert system. Of these, instances of the two leftmost exist today, and the third from the left is underway as software agents using World Wide Web techniques. An instance of the second one from the left (Graphic) is DDT, the topic of the rest of this chapter [D97a].

The Decision Problem

This section presents an example of a decision problem suitable for investigation using the DELTA method. A medium-sized Swedish manufacturing company relied in one of its most important production lines on an old machine, to which spare parts had become increasingly hard to obtain. At a critical moment, the machine broke down in a more serious way than previously. It became clear to management that the machine was a potential danger to future operations unless it was either thoroughly repaired or replaced by a new machine.

A DDT Session

Currently, DDT runs on Windows 95 PCs and Unix workstations, and it is from the latter implementation that this session is taken. When the program is launched without a pre-existing data file, a default decision problem is created. Apart from the traditional **File** menu, the top level menu in DDT consists of the following items:

Settings

- Show hull values
- Utility settings
- Zoom
- Security levels

Evaluations

- Absolute
- Relative set
- Alternative 1
- Alternative 2
- Security check

Table 3.1 Main pop-down menus

NOTE: Due to severe problems with editing and printing this chapter, the PDF reprint will mostly contain the figures (screenshots) from the presentation of the tool. If the chapter is allowed to be more extensive than this reprint, it is not editable in either Word 5.1 or 6.0, neither convertible to a PDF file. Originally, the chapter consisted of pages 47–64 but it is not possible to recreate in its entirety. Only this chapter is affected by these editing problems.

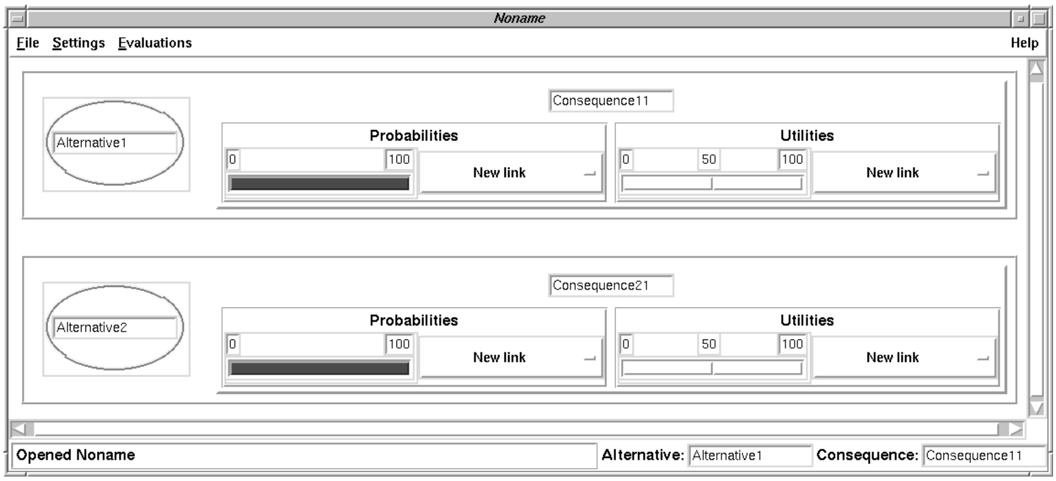

Figure 3.2

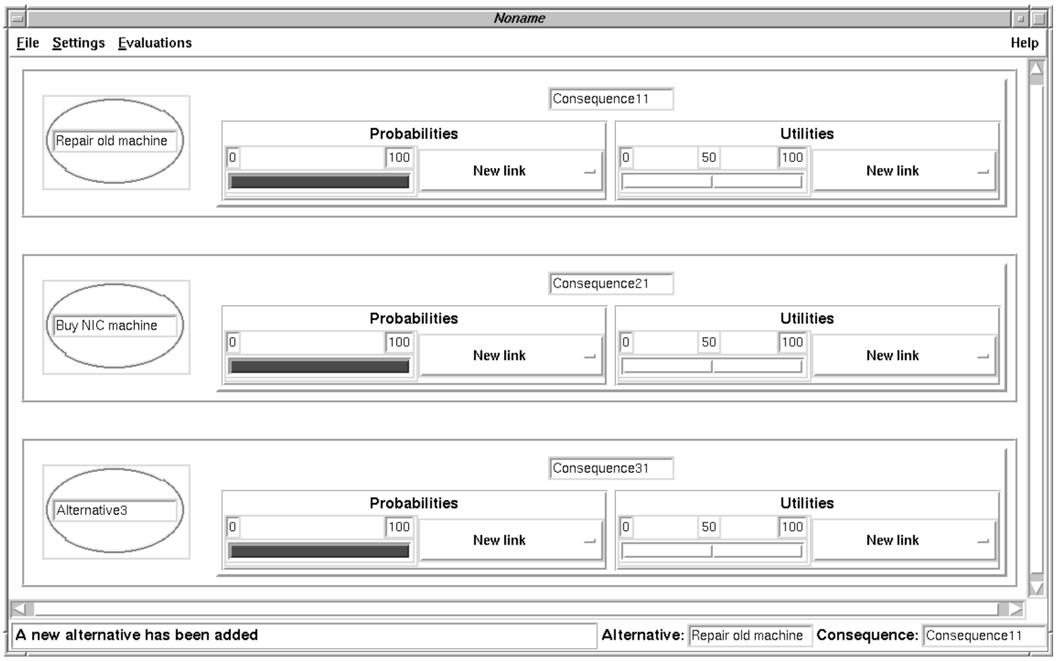

Figure 3.3

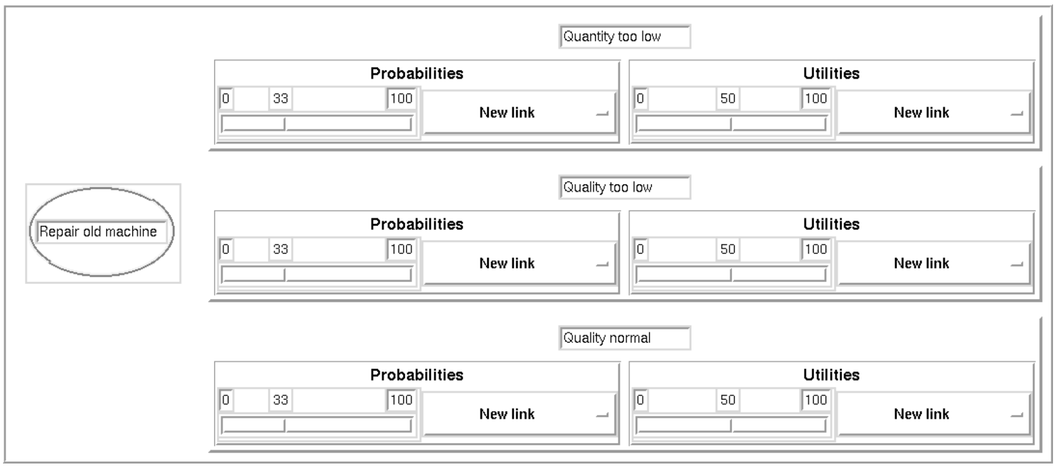

Figure 3.4

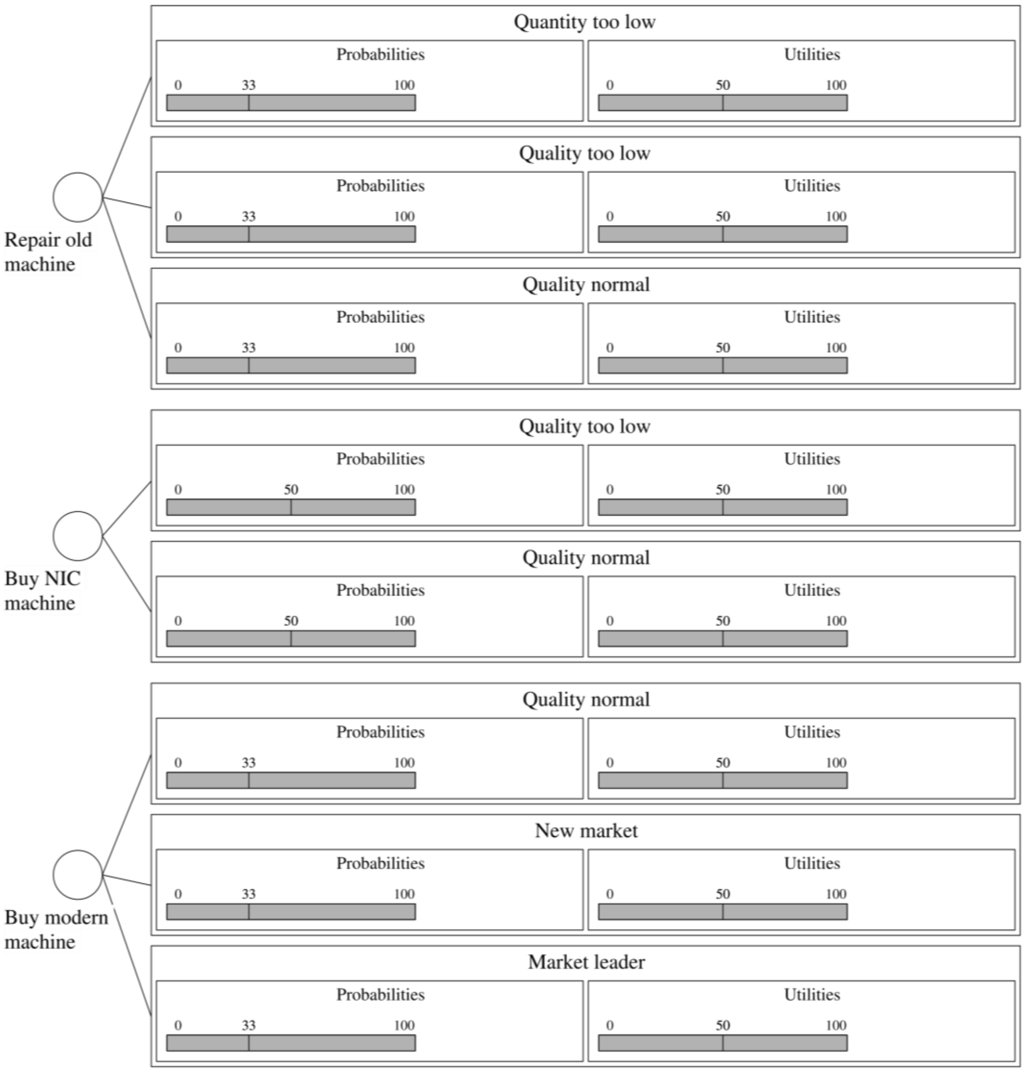

Figure 3.5

User Statements

To begin with, it is assumed that the decision maker is content with the tree and wants to move on to entering probabilities and values. This is done either by dragging the interval endpoints using the mouse or by entering the numbers manually. The interval is modified interactively, and feedback is given if the base is becoming inconsistent as a result of altering an interval. An important difference between probabilities and values is the familiarity among decision makers with [0,1] variables. For probabilities, numbers in the range [0,1] (in the form 0% to 100%) are commonly accepted. For values, on the other hand, the range [0,1] is not the most natural nor the most common. Therefore, as was shown in Figure 3.1, DDT allows any range for the values, even such where greater utility is derived from smaller values, as is the case with for example pollution.

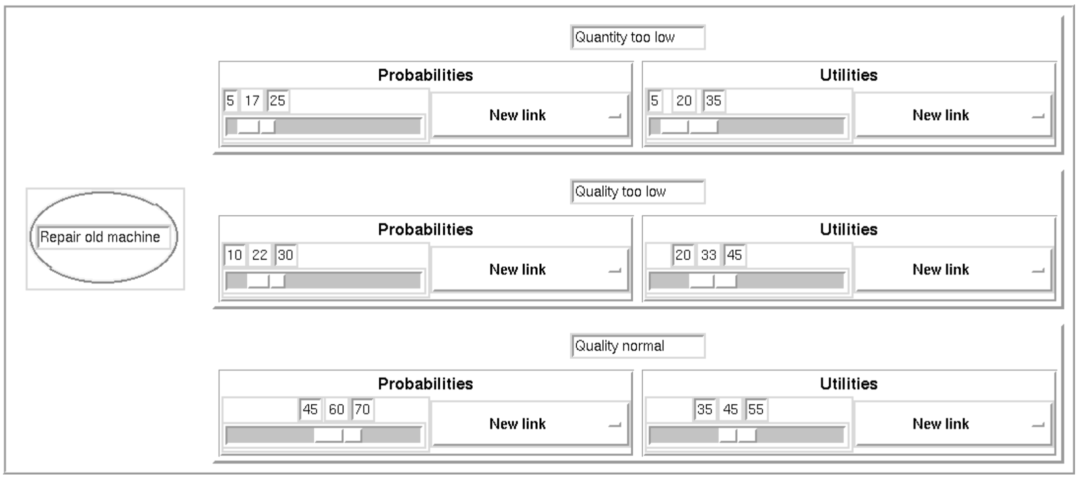

Figure 3.6

A default focal point is suggested by DDT when the decision problem is entered. It can be modified by the decision maker at any time during the evaluation, as long as it is kept consistent. The consistency of the information is maintained by DDT. After the probabilities and values have been entered for the other two alternatives as well, the DELTA decision tree looks like Figure 3.7.

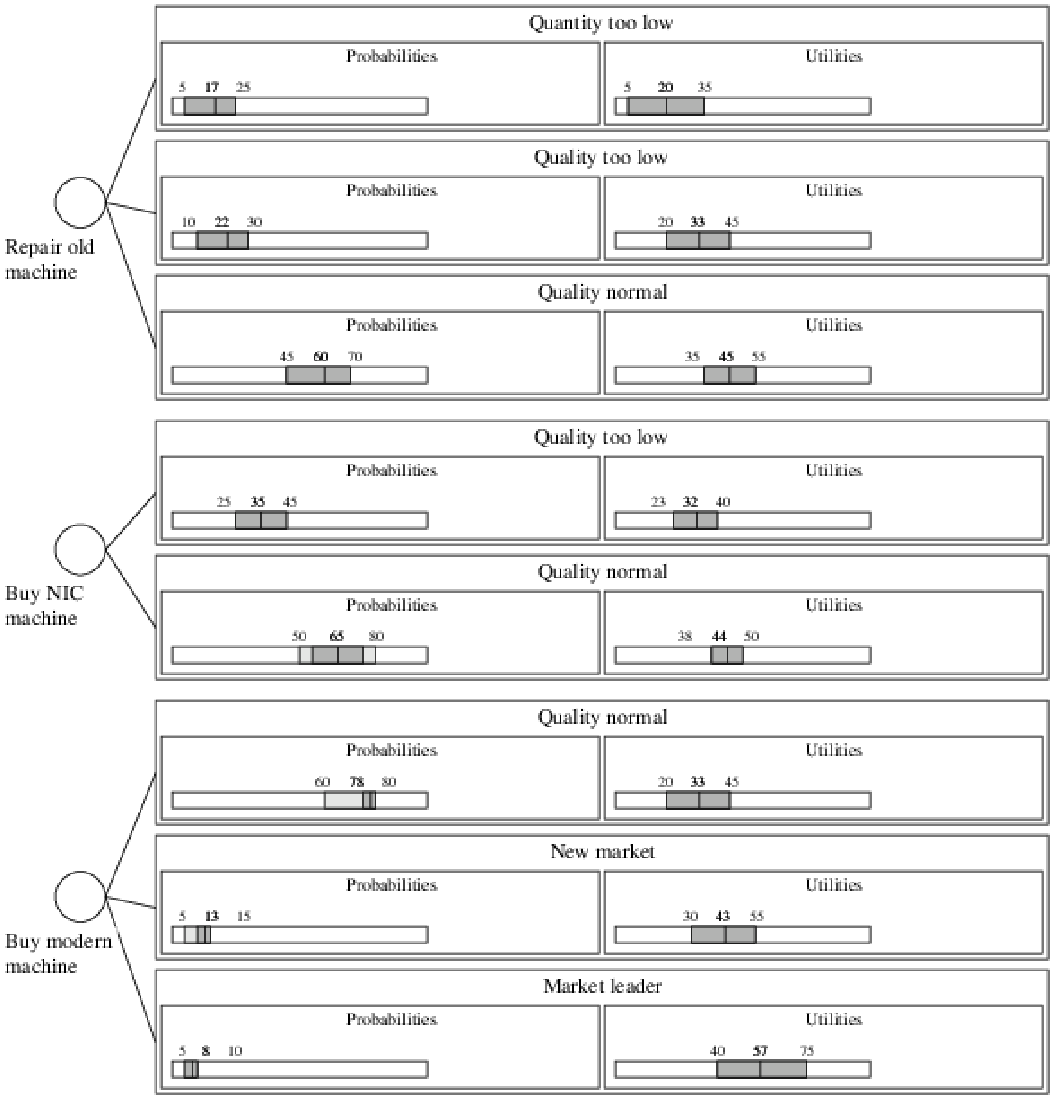

Figure 3.7

Evaluation

Now that all initial information is stored properly in the tree, the evaluation phase can begin. Each evaluation takes place in a separate window, and there may be more than one window active at the same time. In each window, there is a possibility to customise the appearance of the evaluation graphs. The following three pop-down menus are available in the DDT decision analysis tool:

Misc
(see Figure 3.8 below)
Add
(adds a graph to the display)
Delete
(removes a graph from the display)

Table 3.2 Evaluation pop-down menus

In the 'Misc' menu, it is possible to choose which of the maximal, medium, and minimal values are to be shown for the current comparison. In this sample session, it was chosen to compare the alternatives pairwise and then to view the medium differences in the graph. It can be seen in Figure 3.8 that only 'Show mid' is selected and in Figure 3.9 an evaluation mid result for two alternatives is shown.

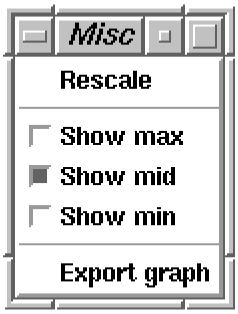

Figure 3.8

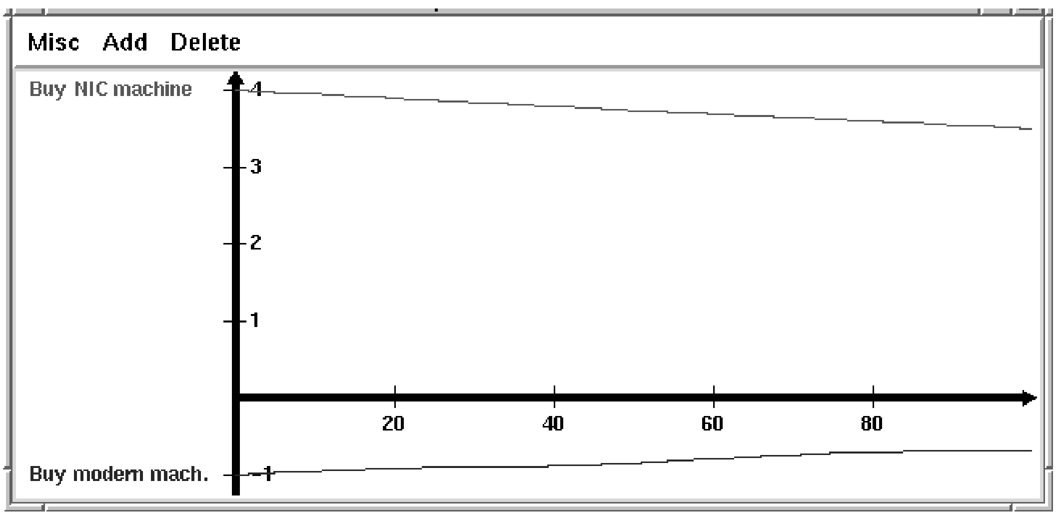

Figure 3.9

4

Representation

In the DELTA method, a decision problem is represented by a *decision frame*. The idea with such a frame is to collect all information necessary for the model in one structure. This structure is then filled in with problem statements. All the probability statements in a decision problem share a common structure because they are all made relative to the same decision frame. They are translated and collected together in a probability base. For value statements, the same is done in a value base. The correspondence between the user model and the representation is summarised in Table 4.1.

User model	Representation
Decision problem	Decision frame
Alternative	Consequence set
Consequence, event	Consequence
Collection of statements	Base
Interval statement	Core interval
	Interval constraint
Range statement	Core interval
	Range constraint
Qualitative statement	Range constraint
Comparative statement	Comparative constraint
Compound statement	Compound constraint
Difference statement	Difference constraint

Table 4.1 Representation of user model

Decision Frames

Chapter 2 contained a discussion on how a decision problem with imprecise data could be modelled. In this chapter, the representation will be considered in more detail. A model in normal form of the situation is created with relevant courses of action and their consequences, should specific events occur. The model is represented by a decision frame. The courses of action are called alternatives in the model, and they are represented by consequence sets in the decision frame. This can be depicted as in Figure 4.1, where the problem is seen to be on alternative–consequence form (AC form), a form of one-level decision tree [J83].

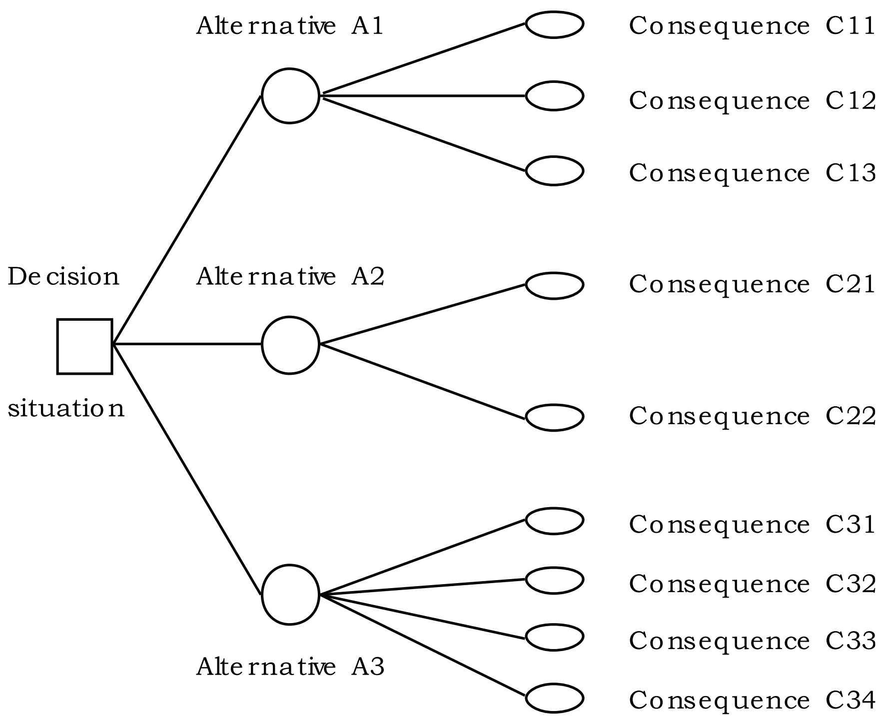

Figure 4.1 A tree view of a decision frame

Following the establishment of a decision frame, the probabilities of the events and the values of the consequences are filled in.[1]

[1] In some presentations, an event is a disjunction of consequences. Here, it is used in a more colloquial way. An event denotes the transition from one state of affairs to

This chapter deals with representing the structure and handling interval statements. Following that, some properties of collections of statements are described.[2] Finally, the translations of probability and value statements are also investigated. The statements themselves and their interpretations are discussed.

Frame Structure

To formalise a decision frame, it is necessary to consider what structure information must be present in order to unambiguously describe a decision problem on AC form. First, a decision frame must capture the structure of the tree. A decision tree consists of sets of consequences. Second, there are statements of probability and value collected in structures called constraint sets and cores. In this chapter, the general interval constraint set will be described first. It can be used both for probability statements and value statements.

In order not to clutter up the definitions with particulars of all components, the terminology clauses introduce generic items used throughout the chapter without further qualification.

> **Terminology:** Given a set X of variables $\{x_i\}_{i \in I}$, the index set I is understood to be $I = \{1,\ldots,n\}$ where n is the number of variables in X.[3]

another. For example, if a dog is run over by a commuter train, then the event is the accident itself and the dog being dead is a consequence of the accident. Still, it will be said that the *consequence* has a probability. This is unambiguous and should not lead to any confusion.

[2] Their computational requirements lead to algorithms for evaluating such properties, one of the topics of Chapter 6.

[3] This is a family $\{x_i\}$ in X, see e.g. [H60].

Constraints

A linear inequality involving a set of variables $\{x_i\}_{i\in I}$ has the form

$$k_1x_1 + k_2x_2 + \ldots + k_nx_n \text{ ¤ } b$$

for some constants k_i, $\forall i\in I$, and b. The relational operator ¤ is any strict or weak inequality such as > or ≤. For the purpose of this chapter the k_i's are often ±1 and are then omitted for the sake of simplicity.[4] Equalities correspond to precise constraints for the respective probabilities and values. In this thesis, however, the interest lies in other kinds of statements, for example interval statements, qualitative statements, and comparative statements. For these statements, interval constraints are used.

> **Definition 4.1:** Given a set of variables $S = \{x_i\}_{i\in I}$, a continuous function $g:S^n\to[0,1]$, and real numbers $a,b \in [0,1]$ with $a \le b$, an *interval constraint* $g(x_1,\ldots,x_n) \in [a,b]$ is a shorter form for a pair of weak inequalities $g(x_1,\ldots,x_n) \ge a$ and $g(x_1,\ldots,x_n) \le b$.

In this manner, both equalities and inequalities can be handled in a uniform way since equalities are represented by intervals [a,a]. There are many types of constraints and they correspond to different types of decision-maker statements as discussed at the end of the chapter.

> **Definition 4.2:** Given a set of variables $\{x_i\}_{i\in I}$ and real numbers $a, b \in [0,1]$ with $a \le b$:
> An *equality constraint* is an interval constraint of the form $x_i \in [a,a]$ where $i \in I$.
> A *range constraint* is an interval constraint of the form $x_i \in [a,b]$ where $i \in I$.
> A *comparative constraint* is an interval constraint of the form $x_i - x_j \in [a,b]$ with $i, j \in I$ and $i \neq j$.

[4] In case of –1, a minus sign is placed directly in front of the k_i's. In the next chapter, other situations are encountered where the k_i's are real numbers in the interval [0,1].

A *difference constraint* is an interval constraint of the form $(x_i - x_j) - (x_k - x_l) \in [a,b]$ with i, j, k, l $\in$ I and $i \neq j \neq k \neq l$.[5]

A *compound constraint* is an interval constraint of the form $x_{h_1} + \ldots + x_{h_m} \in [a,b]$ for $h_1, \ldots, h_m \in I$ and $h_i = h_j$ iff $i = j$.

A *1-constraint* is an interval constraint of the form $k_{h_1}x_{h_1} + k_{h_2}x_{h_2} + \ldots + k_{h_m}x_{h_m} \in [a,b]$ where $k_{h_i} \in \{-1,1\}$ for $h_1, \ldots, h_m \in I$ and $h_i = h_j$ iff $i = j$.

A *linear constraint* is an interval constraint of the form $k_{h_1}x_{h_1} + k_{h_2}x_{h_2} + \ldots + k_{h_m}x_{h_m} \in [a,b]$ where $k_{h_i} \in [0,1]$ for $h_1, \ldots, h_m \in I$ and $h_i = h_j$ iff $i = j$.

This thesis does not explicitly treat non-linear constraints. Thus, in the sequel all interval constraints are linear unless specifically noted. A collection of interval constraints concerning the same set of variables is called a constraint set, and it forms the basis for the representation of decision situations.

Definition 4.3: Given a set of variables $\{x_i\}_{i\in I}$, a *constraint set* in $\{x_i\}_{i\in I}$ is a set of interval constraints in $\{x_i\}_{i\in I}$.

From the definition of an interval constraint, it follows that a constraint set can be seen as a system of inequalities. For a system of inequalities to be meaningful, there must be some vector of variable assignments that satisfies each inequality in the system simultaneously.

Definition 4.4: Given a set of variables $\{x_i\}_{i\in I}$ a *solution* to a system X of inequalities in $\{x_i\}_{i\in I}$ is a real vector $\mathbf{a} = (a_1,\ldots,a_n)$ where each a_i is substituted for x_i such that every inequality in the system is satisfied.[6] The vector **a** is called a *solution vector* to X. The *solution set* for X is {**b** ™ **b** is a solution to X}.

Constraint sets have many properties in common, whether they are probability or value constraint sets. The first question is whether the

[5] Note that this can be written $(x_i + x_l) - (x_j + x_k) \in [a,b]$.

[6] There exists a solution if the substitution of a_i for x_i in X, for all $1 \leq i \leq n$, does not yield a contradiction.

elements in a constraint set are at all compatible with each other. The translates to the problem of whether a constraint set has a solution, i.e. if there exists any vector of real numbers that can be assigned to the variables.

Definition 4.5: Given a set of variables $\{x_i\}_{i \in I}$, a constraint set X in $\{x_i\}_{i \in I}$ is *consistent* **iff** the system of weak inequalities in X has a solution.[7] Otherwise, the constraint set is *inconsistent*. A constraint Z is *consistent with* a constraint set X **iff** the constraint set $\{Z\} \approx X$ is consistent.

In other words, a consistent constraint set is a set where the constraints are at least not contradictory.

Example 4.1: Consider the following constraint set Y in $\{y_i\}_{i \in \{1,2,3,4\}}$:

$y_1 \in [0.30, 0.60]$ $y_1 - y_2 \in [0.10, 0.30]$
$y_2 \in [0.25, 0.55]$ $y_1 - y_3 \in [0.10, 0.40]$
$y_3 \in [0.10, 0.40]$ $y_3 - y_4 \in [-0.10, 0.10]$
$y_4 \in [0.05, 0.20]$ $y_1 + y_2 + y_3 + y_4 \in [1.00, 1.00]$

A solution vector to the system of inequalities that Y represents is (0.40, 0.30, 0.20, 0.10) and thus the constraint set Y is consistent. n

In many of the evaluation algorithms, it is important to find optima for given objective functions. The following definition is intended to simplify the presentation.

Definition 4.6: Given a consistent constraint set X in $\{x_i\}_{i \in I}$ and a function f, ${}^{X}\max(f(x)) =_{def} \sup(a ™ \{f(x) > a\} \approx X$ is consistent). Similarly, ${}^{X}\min(f(x)) =_{def} \inf(a ™ \{f(x) < a\} \approx X$ is consistent)

Example 4.1 (cont'd): Consider the same constraint set Y as above. Let f(y) be $y_1 + y_3$. Then ${}^{Y}\max(y_1 + y_3) = 0.70$ and ${}^{Y}\min(y_1 + y_3) = 0.50$. Those optima are reached in (0.55, 0.25, 0.15, 0.05) and (0.40, 0.30, 0.10, 0.20) respectively. Next let f(y) be $y_1 - y_2 + y_4$ instead. Then ${}^{Y}\max(y_1 - y_2 + y_4) = 0.40$ and ${}^{Y}\min(y_1 - y_2 + y_4) = 0.15$. Those

[7] Then there is a non-empty solution set for X.

optima are reached in (0.45, 0.25, 0.10, 0.20) and
(0.475, 0.375, 0.10, 0.05) respectively. ■

The orthogonal hull is a concept that in each dimension signals which parts are definitely incompatible with the constraint set. The orthogonal hull can be pictured as the result of wrapping the smallest orthogonal hyper-cube around the constraint set.

Definition 4.7: Given a consistent constraint set X in $\{x_i\}_{i\in I}$, the set of pairs $\{\langle {}^X\min(x_i), {}^X\max(x_i)\rangle\}_{i\in I}$ is the *orthogonal hull* of the set and is denoted $\langle {}^X\min(x_i), {}^X\max(x_i)\rangle_n$.

Example 4.1 (cont'd): Consider the same constraint set Y again. Let f(y) be y_1. Then ${}^Y\max(y_1) = 0.55$ and ${}^Y\min(y_1) = 0.35$. Carrying the calculations out for the other three y_i's yields the following hull:
$\{\langle 0.35,0.55\rangle, \langle 0.25,0.375\rangle, \langle 0.10,0.25\rangle, \langle 0.05,0.20\rangle\}$.
Compared to the range constraints in the base
$y_1 \in [0.30, 0.60]$ $y_3 \in [0.10, 0.40]$
$y_2 \in [0.25, 0.55]$ $y_4 \in [0.05, 0.20]$
there are some differences because the comparative constraints do not allow the full ranges to contain consistent points in Y. For example, the upper bound of y_1 has been cut from 0.60 to 0.55. ■

Other hull concepts are possible as well, and in this thesis the symmetric hull is considered. First, a few help definitions are made.

Definition 4.8: Given a constraint set X in $\{x_i\}_{i\in I}$ and the orthogonal hull $H = \langle a_i,b_i\rangle_n$ of X, a *focal point* is a solution vector $(r_1,\ldots,r_n)$ with $a_i \le r_i \le b_i$, $\forall i\in I$. The *hull midpoint* is $(m_1,\ldots,m_n)$ with $m_i = \dfrac{a_i + b_i}{2}$.

Focal points are chosen by the decision-maker according to his appreciation of the decision situation. In DDT, a default focal point is suggested by the tool, and it can be altered as desired as long as it is kept consistent. Next, the standard concept of distance is introduced.

Definition 4.9: Given two vectors **a** and **b**, the *distance function* d is a function that satisfies

(i a) $d(\mathbf{a},\mathbf{b}) > 0$ if $\mathbf{a} \neq \mathbf{b}$

(i b) $d(\mathbf{a},\mathbf{a}) = 0$

(ii) $d(\mathbf{a},\mathbf{b}) = d(\mathbf{b},\mathbf{a})$

(iii) $d(\mathbf{a},\mathbf{b}) \leq d(\mathbf{a},\mathbf{c}) + d(\mathbf{c},\mathbf{b})$ for all **c**.

For the definition to be meaningful in this context, the distance function must be reasonable, even though this does not follow directly from the definition. In many constraint sets, the focal point is not the orthogonal hull midpoint. The hull midpoint need not even be consistent. In those cases, the base is said to be skewed, and the concept of skewness is introduced to describe this.

Definition 4.10: Given a constraint set X in $\{x_i\}_{i\in I}$, two real vectors $\mathbf{a} = (a_1,\ldots,a_n)$ and $\mathbf{b} = (b_1,\ldots,b_n)$ of the orthogonal hull $\langle a_i,b_i\rangle_n$ of X, a distance function d, a constant $k \in [0,1]$, a hull midpoint **m**, and a focal point **r**. The *skewness* of the base X with respect to **r** is $k \cdot \frac{d(\mathbf{r},\mathbf{m})}{d(\mathbf{a},\mathbf{b})}$.

When a base is skewed, there exists a way of avoiding this asymmetry by using the symmetric hull instead.

Definition 4.11: Given a constraint set X in $\{x_i\}_{i\in I}$, the orthogonal hull $\langle a_i,b_i\rangle_n$ of X, and a focal point $(r_1,\ldots,r_n)$. Let $d_i = \min(r_i-a_i, b_i-r_i)$, $\forall i\in I$. The *symmetric hull* is $\langle r_i-d_i,r_i+d_i\rangle_n$.

Example 4.1 (cont'd): Consider the same constraint set Y again. Let r = (0.42, 0.29, 0.17, 0.12) be a focal point. Carrying the calculations out for the four y_i's yields the following hull:
$\{\langle 0.35,0.49\rangle, \langle 0.25,0.33\rangle, \langle 0.10,0.24\rangle, \langle 0.05,0.19\rangle\}$.
Compared to the orthogonal hull of the base
$\{\langle 0.35,0.55\rangle, \langle 0.25,0.375\rangle, \langle 0.10,0.25\rangle, \langle 0.05,0.20\rangle\}$

there are some differences because the focal point is not the midpoint of the orthogonal hull.[8] ∎

Note: If the symmetric hull coincides with the orthogonal hull, then the skewness is zero. This follows from d(**r**,**m**) = 0 if the midpoint **m** is equal to the focal point **r**.

The generic term *hull* will be used for the orthogonal hull or the symmetric hull as appropriate.

Bases

A base consists of a constraint set for a set of variables together with a core. Constraints and core intervals have different roles in specifying a decision situation. The constraints represent "negative" information, which vectors are not part of the solution sets. The contents of constraints specify which ranges are infeasible by excluding them from the solutions. This is in contrast to core intervals, which represent "positive" information in the sense that the decision-maker enters information about sub-intervals that are felt to be the most central ones and that no further discrimination is possible within those ranges.

Definition 4.12: Given a constraint set X in $\{x_i\}_{i\in I}$ and the orthogonal hull $\langle a_i,b_i\rangle_n$ of X, a *core interval* of x_i is an interval $[c_i,d_i]$ such that $a_i \le c_i \le d_i \le b_i$. A *core* $[c_i,d_i]_n$ of $\{x_i\}_{i\in I}$ is a set of core intervals $\{[c_i,d_i]\}_{i\in I}$, one for each x_i.

As for constraint sets, the core might not be meaningful in the sense that it may contain no possible variable assignments able to satisfy all the inequalities. This is quite similar to the concept of consistency for constraint sets, but for core intervals, the requirement is slightly different. It is required that the focal point is contained within the core.

[8] Note that the symmetric hull is always tighter since the upper hull value is decreased or the lower increased. Also note that only one of the upper and lower values is changed for each index.

Definition 4.13: Given a consistent constraint set X in $\{x_i\}_{i\in I}$ and a focal point $\mathbf{r} = (r_1,\ldots,r_n)$, the core $[c_i,d_i]_n$ of $\{x_i\}_{i\in I}$ is *permitted with respect to* $\boldsymbol{r}$ **iff** $c_i \le r_i \le d_i$, $\forall i\in I$.

Example 4.1 (cont'd): Consider the same constraint set Y again. Recall that the constraint set is
$y_1 \in [0.30, 0.60]$ $y_3 \in [0.10, 0.40]$
$y_2 \in [0.25, 0.55]$ $y_4 \in [0.05, 0.20]$
and that $\mathbf{r} = (r_1,\ldots,r_4) = (0.42, 0.29, 0.17, 0.12)$ is the focal point.
Let the core be
$y_1 \in [0.40, 0.45]$ $y_3 \in [0.15, 0.20]$
$y_2 \in [0.25, 0.35]$ $y_4 \in [0.10, 0.15]$.
Now r_1 is contained in the core interval of y_1, and the same is true for the other three y_i's. Thus the core is permitted. The interpretation of, for example, the information about y_1 is that, according to the decision-maker, the value of y_1 is not below 0.30 and not above 0.60. In addition, the most plausible values for y_1 are between 0.40 and 0.45. The single most representative value is 0.42, but the DELTA method tries not to exploit the latter fact if not necessary. ■

A base is simply a collection of constraints and the core that belongs to the variables in the set. The idea with a base is to represent a class of functions over a finite, discrete set of consequences.

Definition 4.14: Given a set $\{x_i\}_{i\in I}$ of variables and a focal point **r**, a *base* X in $\{x_i\}_{i\in I}$ consists of a constraint set X_C in $\{x_i\}_{i\in I}$ and a core X_K of $\{x_i\}_{i\in I}$. The base X is *consistent* if X_C is consistent and X_K is permitted with respect to **r**.

It is natural to consider values near the boundaries of the intervals in a constraint set as being less reliable than more central values, due to interval constraints being deliberately imprecise. The core, on the other hand, represents the most reliable estimates. It is therefore desirable to be able to study the core with varying degrees of expansion, i.e. studying smaller or larger extensions to the original core. The expansion can be regarded as a focus parameter that zooms out from central sub-intervals to the full constraint intervals. It is not a measure of volume but rather of the strength of statements as volume is added to the original core. Conversely, if the core itself is not enough to yield the desired evaluation results, it can be contracted towards the focal point with varying degrees of contraction.

Definition 4.15: Given a base X in $\{x_i\}_{i\in I}$, a set of real numbers $\{a_i,b_i\}_{i\in I}$, a core $[c_i,d_i]_n$ of $\{x_i\}_{i\in I}$, and a real number $\pi \in [0,1]$, a *π-flation* of X is to replace the core by $[c_i+\pi\cdot(a_i-c_i),\ d_i+\pi\cdot(b_i-d_i)]_n$. If the set $\{a_i,b_i\}_{i\in I}$ is the hull $\langle a_i,b_i\rangle_n$ then it is called a *π-expansion* of X.[9] If $(r_1,\ldots,r_n)$ is a focal point and $a_i = b_i = r_i$, then it is called a *π-contraction* of X.

The π-flation is a linear procedure, but non-linear procedures are plausible as well. In addition, the procedure can work from either side ((L)π-flation and (R)π-flation) or with varying, even non-uniform rates of expansion or contraction.[10]

Probability Bases

A probability base contains a collection of probability statements in the form of constraints and a core.

Definition 4.16: Given a set $\{C_{ik}\}_{k\in K}$ of disjoint and exhaustive consequences, a base P in $\{p_{ik}\}_{k\in K}$, $K = \{1,\ldots,m_i\}$, and a discrete, finite probability mass function $\prod:C\rightarrow[0,1]$ over $\{C_{ik}\}$. Let p_{ik} denote the function value $\prod(C_{ik})$. $\prod$ obeys the standard probability axioms, and thus $p_{ik} \in [0,1]$ and $\sum_k p_{ik} = 1$ are default constraints in the constraint set P_C. Then P is a *probability base.*

Thus, a probability base can be seen as characterising a set of discrete probability distributions.[11] The core P_K can be thought of as an attempt to estimate a class of mass functions by estimating the individual discrete function values.

[9] Note that $a_i \leq c_i \leq d_i \leq b_i$.

[10] For simplicity, especially in calculated examples, only the linear expansions and contractions are employed in this thesis.

[11] See for example [WP90] for a discussion of similar ideas.

The collection of (translated) probability constraints can be categorised into different types of bases progressively harder to evaluate and differing more and more from the standard equality case. Using the categories of constraints from earlier in the chapter, a hierarchy can be defined.

> **Definition 4.17:** A probability base is of *type* P_0 (a P_0-base for short) if all constraints are equality constraints plus one compound constraint (the normalisation) for each consequence set.
> A probability base is of *type* P_1 (a P_1-base for short) if the constraints are range constraints plus one compound constraint (the normalisation) for each consequence set.
> A probability base is of *type* P_2 (a P_2-base for short) if the constraints are range or comparative constraints plus one compound constraint (the normalisation) for each consequence set.
> A probability base is of *type* P_3 (a P_3-base for short) if the constraints are linear constraints.
> A probability base is of *type* P_4 (a P_4-base for short) if the constraints are non-linear constraints.

P_0-bases correspond to the standard models discussed in Chapter 1. A P_1-base corresponds to the simplest case of generalised bases containing only interval statements. This is the most common generalisation, encountered in for example [WP90]. Sometimes a P_1-base is called an interval base. A P_2-base is an interval base extended with comparisons between probabilities. A P_3-base contains all constraints plausible within this framework. P_4-bases may contain non-linear constraints and are beyond the scope of this thesis, but are included for completeness.

Example 4.2: Consider a probability base P with a constraint set
$p_{11} \in [0.15, 0.30]$ $p_{13} \in [0.40, 0.55]$
$p_{12} \in [0.20, 0.30]$ $p_{14} \in [0.10, 0.15]$
$p_{11} + p_{12} + p_{13} + p_{14} \in [1,1]$.

This base is of type P_1 because all constraints are range constraints except for the normalisation. If the comparative constraint
$p_{13} - p_{12} \in [0.15,0.35]$
is added to the base, it will be of type P_2.
Further, if the compound constraint
$p_{12} + p_{14} \in [0.25,0.40]$
is added to the base, it will become a base of type P_3.

Value Bases

Requirements similar to those for probability variables can be found for value variables. There are apparent similarities between probability and value statements but there are differences as well. The normalisation ($\sum_k p_{ik} = 1$) requires the probability variables of a set of exhaustive and mutually exclusive consequences to sum to one. No such dimension-reducing constraint exists for the value variables.

Definition 4.18: Given a set $\{C_{ik}\}_{k\in K}$ of disjoint and exhaustive consequences, a base V in $\{v_{ik}\}_{k\in K}$, $K = \{1,\ldots,m_i\}$, and a discrete, finite value function Ω:C→[0,1]. Let v_{ik} denote the function value $\Omega(C_{ik})$. Because of the range of Ω, $v_{ik} \in [0,1]$ are default constraints in the constraint set V_C. Then V is a *value base*.

Similar to probability bases, a value base can be seen as characterising a set of value functions. The value core V_K can be seen as an attempt to estimate a class of value functions. The collection of (translated) value constraints can also be categorised into a hierarchy of bases.

Definition 4.19: A value base is of *type V_0* (a V_0-base for short) if all constraints are equality constraints.
A value base is of *type V_1* (a V_1-base for short) if all constraints are range constraints.
A value base is of *type V_2* (a V_2-base for short) if the constraints are range or comparative constraints.
A value base is of *type V_3* (a V_3-base for short) if the constraints are linear constraints.

A value base is of *type V_4* (a V_4-base for short) if the constraints are non-linear constraints.

V_0-bases correspond to the standard models discussed in Chapter 1. A V_1-base corresponds to the simplest case of generalised bases containing only interval statements. This is the most common generalisation. A V_2-base is an interval value base extended with comparisons between values of any alternatives. The alternatives might be dependent. A V_3-base contains all constraints plausible within this framework. In practice, this includes differences in particular, but not compound value statements since they lack semantic content. A V_4-base may contain any interval constraints, even non-linear. Those are beyond the scope of this thesis but are included for completeness and future reference.

Example 4.3: Consider a value base V with a constraint set
$v_{11} \in [0.40,0.55]$ $v_{13} \in [0.05,0.15]$
$v_{12} \in [0.30,0.65]$ $v_{14} \in [0.80,0.95]$
This base is of type V_1 because all constraints are range constraints. If the comparative constraint $v_{12} - v_{11} \in [0.00,0.15]$ is added to the base, it will be of type V_2. Further, if the difference constraint $(v_{14} - v_{12}) - (v_{11} - v_{12}) \in [0.05,0.20]$ is added to the base, it will become a base of type V_3.

Frames

Using the above concepts of consequence, constraint, core, and base, it is possible to model the decision-maker's situation in a decision frame.

Definition 4.20: Given a decision situation with m alternatives $(A_1,\ldots,A_m)$, each with m_i consequences, and statements about the probabilities and values of those consequences. A *decision frame* is a structure $\langle C,P,V \rangle = \langle \{\{C_{ik}\}_{m_i}\}_m,P,V \rangle$ containing the following representation of the situation:

- For each alternative A_i the corresponding consequence set $\{C_{ik}\}_{k\in K_i}$ for $K_i = \{1,\ldots,m_i\}$.

- A probability base P containing all probability statements in the form of constraints and a core.
- A value base V containing all value statements in the form of constraints and a core.

Compare the decision frame to Table 4.1 at the beginning of the chapter. The frame captures a decision problem in AC form, a one-level tree problem in normal form. As problems on other forms can be converted to this form, it is a general structure, highly applicable to a wide range of problems. The frame is also the key data structure in the DELTALIB implementation, holding references to other structure information and to the bases containing most of the information. To simplify the presentation in the following chapters, a shorthand notation for frames with bases of specific types is introduced.

> **Terminology:** When a decision frame $\langle C,P_i,V_j\rangle$ is referred to with indices on the bases, it should be interpreted as a decision frame containing a probability base of type P_i and a value base of type V_j.[12]

Translations

User statements can be translated into constraints and core intervals in various ways. One technique is to present the decision-maker with a numeric interface where the statements can be interactively entered. For human decision-makers, this can be done as in the DDT tool presented in Chapter 3. There, statements are entered by manipulating rulers representing constraints, core intervals, and focal points. Another technique is to translate linguistic statements by translation rules. While the objective is to preserve as much as possible of the original meaning of each statement, the nature of translation rules is necessarily approximate because of the ambiguity inherent in linguistic statements.

[12] For example, the DDT software currently handles $\langle C,P_2,V_2\rangle$ frames because of the availability of fast algorithms (see Chapter 6).

Still, it is an important way of entering information into a decision frame. The last sections of this chapter consider the translation of probability and value statements respectively.

Probability Translations

To handle linguistic probability statements computationally, they must be translated into a suitable form. This means that they are translated into inequalities, using the above suggested interval form. The following translations are proposed, which are by no means the only ones possible.

Range Statements

Range statements are of two types, those that translate into core intervals and those that translate into range constraints. While both types are statements about the probabilities of single consequences, their semantical contents differ. Statements translated into core intervals are of a "positive" type in the sense that they express an interval within which the decision-maker is unable to further discriminate between different numbers. On the other hand, statements translated into range constraints are of a "negative" type in the sense that they originate from estimates where the decision-maker has been deliberately imprecise and included numbers that are less likely but not entirely unlikely. Thus, a reasonable interpretation of such statements is that the estimated number is not outside of the given interval but without any explicit hint as to where it might be inside of it.

Statement: *The probability of C_{11} is m.*
Translation: The core interval $p_{11} \in [m-\varepsilon_{11}, m+\varepsilon_{11}]$
Comment: The constant ε_{11} is some small number expressing the uncertainty in seemingly categorical statements.

Statement: *The probability of C_{12} is about m.*
Translation: The core interval $p_{12} \in [m-\eta_{12}, m+\eta_{12}]$

Comment: η_{12} is some rather small number, though considerably larger than ε_{11} above. It expresses greater uncertainty as indicated by the choice of words.

Statement: *The probability of C_{13} is definitely between a_{13} and b_{13}.*

Translation: The range constraint $p_{13} \in [a_{13}, b_{13}]$

Comment: The statement can be taken literally, as is done in the proposed translation, or it can be modified with small constants as suggested in the previous cases. The latter way is more appropriate when a_{13} and b_{13} are close to each other.

Statement: *The probability of C_{14} is greater than m.*

Translation: The range constraint $p_{14} \in [m+\eta_{14}, m+\lambda_{14}]$

Comment: The more obvious and direct interpretation of $p_{14} > m$ is not advisable. If it were to be used, the translation would be something like $p_{14} > m \Leftrightarrow p_{14} \in [m+\varepsilon_{14}, 1]$, which is a misinterpretation of both limits. For the lower limit, it is quite possible that the decision-maker does not mean that p_{14} exceeds m by a very small, hardly noticeable amount. Thus, η_{14} is some rather small number, though considerably larger than ε_{11} above. For the upper limit, the expression "greater than" is often used to denote a noticeable but not extremely large difference between p_{14} and m. If the difference is perceived to be gigantic, words such as "much greater than" should be used.

Statement: *The probability of C_{15} is much greater than m.*

Translation: The range constraint $p_{15} \in [m+\beta_{15}, m+\theta_{15}]$

Comment: The direct translation $p_{15} > m+\beta_{15}$ for some reasonable large β_{15} is again not advisable for much the same reasons as above. The expression "much greater than" is often used to denote a sizeable difference between p_{15} and m.

Statement: *The probability of C_{16} is less than m.*

Translation: The range constraint $p_{16} \in [m-\lambda_{16}, m-\eta_{16}]$

Comment: The translation $p_{16} < m$ is not recommended, since it would mean $p_{16} < m \Leftrightarrow p_{16} \in [0, m-\varepsilon_{16}]$, which is a misinterpretation of both limits. For the lower limit, the expression "less than" is often used to denote a noticeable but not extremely large difference between p_{16} and m. It is almost never meant that C_{16} is totally impossible, i.e. that p_{16} could assume the value zero. For the upper limit, the reasons are as above for $m-\lambda_{16} \geq 0$.

Statement: *The probability of C_{17} is much less than m.*

Translation: The range constraint $p_{17} \in [m-\theta_{17}, m-\beta_{17}]$

Comment: The direct translation $p_{17} < m-\beta_{17}$ for some reasonable large β_{17} is not advisable, for the same reasons as above. θ_{17} is a suitable, larger constant such that $m-\theta_{17} \geq 0$.

Qualitative Statements

Qualitative statements are translated into range constraints (except for compound statements, see below). They are of a "negative" type in the same sense as above and are all translated into constraints.

Statement: *The event C_{21} is probable.*

Translation: The range constraint $p_{21} \in [a_{21}, b_{21}]$

Comment: a_{21} and b_{21} are constants suitable for the situation. Since the statement expresses some confidence in the event, a_{21} is a rather high value. b_{21} can be 1, but this is not necessary, and often not even appropriate.

Statement: *The event C_{22} is possible.*

Translation: The range constraint $p_{22} \in [a_{22}, b_{22}]$

Comment: a_{22} and b_{22} are constants convenient for the situation. a_{22} is obviously lower than a_{21} above since the decision-maker expresses considerably less confidence in the event occurring than in the previous statement. In the same manner, b_{22} is less than b_{21}.

Statement: *The event C_{23} is improbable.*

Translation: The range constraint $p_{23} \in [a_{23}, b_{23}]$

Comment: a_{23} is rather small but not zero, since that would mean the event could be altogether impossible. The upper limit b_{23} is lower than the other upper limits above, but need not necessarily be very close to zero, since the word improbable used in its colloquial meaning can imply some fair probability of the event actually occurring.

Comparative Statements

A comparative statement compares the probabilities of two consequences occurring with one another.

Statement: *The probability of C_{31} is equal to the probability of C_{32}.*
Translation: The comparative constraint $p_{31} - p_{32} \in [-\varepsilon_{31}, +\varepsilon_{31}]$
Comment: ε_{31} is some small number expressing the uncertainty in categorical statements. Note that this interval contains negative as well as positive values.

Statement: *The probability of C_{33} is about equal to the probability of C_{34}.*
Translation: The comparative constraint $p_{33} - p_{34} \in [-\eta_{33}, +\eta_{33}]$
Comment: η_{33} is some rather small number, though considerably larger than ε_{31} above. It expresses the greater uncertainty as indicated by the choice of words.

Statement: *The event C_{35} is more probable than C_{36}.*
Translation: The comparative constraint $p_{35} - p_{36} \in [\eta_{35}, \lambda_{35}]$
Comment: The translation $p_{35} > p_{36}$ is not advisable, because then, the meaning would be $p_{35} > p_{36} \Leftrightarrow p_{35} - p_{36} \in [\varepsilon, 1]$, which, as in the single statement above, is a misinterpretation of both limits. If the difference is perceived to be very large, an expression such as "much more probable than" should be used.

Statement: *The event C_{37} is much more probable than C_{38}.*
Translation: The comparative constraint $p_{37} - p_{38} \in [\beta_{37}, \theta_{37}]$
Comment: Cf. similar translations above.

Statement: *The event C_{39} is less probable than C_{40}.*
Translation: The comparative constraint $p_{39} - p_{40} \in [-\lambda_{39}, -\eta_{39}]$
Comment: Cf. similar translations above.

Statement: *The event C_{41} is much less probable than C_{42}.*
Translation: The comparative constraint $p_{41} - p_{42} \in [-\lambda_{41}, -\eta_{41}]$
Comment: Cf. similar translations above.

Compound Statements

It is sometimes more flexible not to be forced to make probability statements about each individual consequence, should that information not be available, too expensive to collect, or too unreliable. It would be more expressive to be able to use either information about a group of consequences or about the individual components, or both. The representation allows for compound statements of probability, i.e. statements such as *the probability of C_{51}, C_{52}, or C_{53} occurring is* ___ for any of the range or qualitative probability statements described above. Instead of the full translations of all the possible compound probability statements, only a few examples are given.

Statement: *The probability of C_{51}, C_{52}, or C_{53} is about m.*
Translation: The constraint $p_{51}+p_{52}+p_{53} \in [m-\eta_{123}, m+\eta_{123}]$
Statement: *The probability of C_{54}, C_{55}, or C_{56} is between a_{456} and b_{456}.*
Translation: The constraint $p_{54}+p_{55}+p_{56} \in [a_{456}, b_{456}]$

In the case of compound probability statements, there is no requirement for probabilities to be given for all single consequences in a compound statement but there is often more information to gain from actually specifying them. This is left to the decision-maker's discretion.

Difference Statements

A difference statement compares the difference between the probabilities of two consequences occurring with the difference between the probabilities of two others.

Statement: *The difference in probability between C_{61} and C_{62} is equal to the difference in probability between C_{63} and C_{64}.*
Translation: The constraint $(p_{61}-p_{62}) - (p_{63}-p_{64}) \in [-\varepsilon_{61}, \varepsilon_{61}]$
Comment: ε_{61} is again some small number expressing the uncertainty in categorical statements.

Statement: *The difference in probability between C_{65} and C_{66} is about equal to the difference in probability between C_{67} and C_{68}.*
Translation: The constraint $(p_{65}–p_{66}) – (p_{67}–p_{68}) \in [–\eta_{65}, \eta_{65}]$

Statement: *The difference in probability between C_{69} and C_{70} is greater than the difference in probability between C_{71} and C_{72}.*
Translation: The constraint $(p_{69}–p_{70}) – (p_{71}–p_{72}) \in [\eta_{69}, \lambda_{69}]$

Statement: *The difference in probability between C_{73} and C_{74} is much greater than the difference in probability between C_{75} and C_{76}.*
Translation: The constraint $(p_{73}–p_{74}) – (p_{75}–p_{76}) \in [\beta_{73}, \theta_{73}]$

Value Translations

Value statements are considered in a manner similar to the probability statements. The value statements need to be translated into interval form in order to be entered into the decision frame. Again, one objective is to preserve as much as possible of the original meaning of each statement. The translations are only proposals and other ones are equally possible for the DELTA method.[13]

Range Statements

Statement: *The value of C_{11} is v.*
Translation: The core interval $v_{11} \in [v–\varepsilon_{11}, v+\varepsilon_{11}]$

Statement: *The value of C_{12} is about v.*
Translation: The core interval $v_{12} \in [v–\eta_{12}, v+\eta_{12}]$

Statement: *The value of C_{13} is definitely between a_{13} and b_{13}.*
Translation: The range constraint $v_{13} \in [a_{13}, b_{13}]$

Statement: *The value of C_{14} is greater than v.*
Translation: The range constraint $v_{14} \in [v+\eta_{14}, v+\lambda_{14}]$

Statement: *The value of C_{15} is much greater than v.*
Translation: The range constraint $v_{15} \in [v+\beta_{15}, v+\theta_{15}]$

[13] Comments on the translations are left out since they would be similar to those for the probability translations above.

Statement: *The value of C_{16} is less than v.*
Translation: The range constraint $v_{16} \in [v-\lambda_{16}, v-\eta_{16}]$

Statement: *The value of C_{17} is much less than v.*
Translation: The range constraint $v_{17} \in [v-\theta_{17}, v-\beta_{17}]$

Qualitative Statements

Statement: *The event C_{21} is desirable.*
Translation: The range constraint $v_{21} \in [a_{21}, b_{21}]$

Statement: *The event C_{22} is acceptable.*
Translation: The range constraint $v_{22} \in [a_{22}, b_{22}]$

Statement: *The event C_{23} is undesirable.*
Translation: The range constraint $v_{23} \in [a_{23}, b_{23}]$

Comparative Statements

Statement: *The events C_{31} and C_{32} are as desirable.*
Translation: The comparative constraint $v_{31} - v_{32} \in [-\varepsilon_{31}, \varepsilon_{31}]$

Statement: *The events C_{33} and C_{34} are about as desirable.*
Translation: The comparative constraint $v_{33} - v_{34} \in [-\eta_{33}, \eta_{33}]$

Statement: *The event C_{35} is more desirable than C_{36}.*
Translation: The comparative constraint $v_{35} - v_{36} \in [\eta_{35}, \lambda_{35}]$

Statement: *The event C_{37} is much more desirable than C_{38}.*
Translation: The comparative constraint $v_{37} - v_{38} \in [\beta_{37}, \theta_{37}]$

Statement: *The event C_{39} is less desirable than C_{40}.*
Translation: The comparative constraint $v_{39} - v_{40} \in [-\lambda_{39}, -\eta_{39}]$

Statement: *The event C_{41} is much less desirable than C_{42}.*
Translation: The comparative constraint $v_{41} - v_{42} \in [-\lambda_{41}, -\eta_{41}]$

Compound Statements

There are no translations suggested for compound value statements. The specification of values should be on a per-consequence basis. If it is not possible to separate the outcomes of several events, they ought to be modelled as a single event instead.

Difference Statements

Statement: *The difference in value between C_{61} and C_{62} is equal to the difference in value between C_{63} and C_{64}.*

Translation: The constraint $(v_{61}–v_{62}) – (v_{63}–v_{64}) \in [–\varepsilon_{61}, \varepsilon_{61}]$

Statement: *The difference in value between C_{65} and C_{66} is about equal to the difference in value between C_{67} and C_{68}.*

Translation: The constraint $(v_{65}–v_{66})–(v_{67}–v_{68}) \in [–\eta_{65}, \eta_{65}]$

Statement: *The difference in value between C_{69} and C_{70} is greater than the difference in value between C_{71} and C_{72}.*

Translation: The constraint $(v_{69}–v_{70}) – (v_{71}–v_{72}) \in [\eta_{69}, \lambda_{69}]$

Statement: *The difference in value between C_{73} and C_{74} is much greater than the difference in value between C_{75} and C_{76}.*

Translation: The constraint $(v_{73}–v_{74}) – (v_{75}–v_{76}) \in [\beta_{73}, \theta_{73}]$

Child in a chair, Sunday night
Listens in the kitchen's yellow light
Child in a chair, small and still
Elbow on the window's dusty sill
Cheek on a window cool as glass
Waiting for the painted night to pass

Child in a chair, Sunday night
Listens in the kitchen's yellow light
Faint and faded stars arrive
Moving like a movie on the sky
Child never dreams of what might have been
Believes the evening is meant for him

I was a child on a Sunday night
Hearing the wind, talking to the land
And letting the time slip through my hands

– P. Ivers

5

Evaluation

This chapter on evaluation is divided into three sections. The first section, Evaluation Rules, discusses the expected value rule and a number of proposed rules to either replace or supplement it. The rules are discussed from a choice rather than preference view. In the next section, DELTA dominance is introduced as a unifying concept for many of the dominance rules in current use. In both of these sections, all rules are discussed relative to a special decision frame with only equality constraints in order to simplify the presentation. The last section, Frame Evaluation, again considers complete decision frames with all kinds of interval constraints, making the selection procedures more complicated as imprecision enters into the evaluation. The terminology clauses introduce generic terms used throughout the chapter without further qualification or explanation. The idea is to make the text lighter in order to facilitate a read flow by concentrating some of the introduction of terminology to the beginning and then using the terms without needing to introduce them in every definition.

> **Terminology:** Given a decision frame ⟨C,P,V⟩, the functions f, g, and h are specified as $f:\mathbf{R}^i \to [0,1]$, $g:\mathbf{R}^j \to [0,1]$, and $h:\mathbf{R}^k \to [0,1]$ with $i, j, k \in N+$ as appropriate. The α and β parameters are real numbers in the range [0,1].

Evaluation Rules

The special kind of decision frame that will be used in this and the following section is the e-frame, similar to the ordinary complete frame but allowing only constraints of the equality type, thus postponing problems of imprecision to the last section when appropriate evaluation rules have been established.

Terminology: A decision frame $\langle C,P_0,V_0\rangle$ is called an *e-frame* (e for equality) since all interval constraints in P_0 and V_0 are equality constraints (except the normalisations in P_0).

The Expected Value Rule

A large group of evaluation functions is the family of all functions that assign a numerical value to a consequence set for subsequent comparison. Such an evaluation function results in numeric values ranking the alternatives (or more precisely, the consequence sets).

Definition 5.1: Given a decision e-frame $\langle\{\{C_{ik}\}_{m_i}\}_m,P_0,V_0\rangle$ and a function f, the *numeric value* $N(C_i)$ of a consequence set $\{C_{ik}\}_{m_i}$ is a function $f(p_{i1},\ldots,p_{im_i},v_{i1},\ldots,v_{im_i})$ over all consequences C_{ik} in the consequence set.

To be reasonable, the value of $N(C_i)$ should range over the interval [0,1] since the values range over that interval.

Example 5.1: Consider a decision situation involving a number of consequence sets of which C_1 has three consequences. The decision e-frame contains the following data:

$p_{11} = 0.35$
$p_{12} = 0.45$
$p_{13} = 0.20$
$v_{11} = 0.20$
$v_{12} = 0.55$
$v_{13} = 0.80$

Assume that the numeric value $N(C_i)$ is given by the function $\sum_k (p_{ik})^2 \cdot v_{ik}$. Then the numeric value $N(C_1)$ becomes $(0.35)^2 \cdot 0.20 + (0.45)^2 \cdot 0.55 + (0.20)^2 \cdot 0.80 = 0.167875$. ■

As was indicated already in Chapter 1, the expected value seems to be one of the more natural rules to apply to a decision problem on AC form. This might partly be because the expected value $E(C_i)$ is established in the area of mathematical statistics, where it is employed as the "mean" value to be assigned to a stochastic variable taking on various values with specific probabilities. $E(C_i)$ is an instance of $N(C_i)$ above. In this thesis, only discrete probability distributions are considered, and thus the following definition of the expected value applies.

Definition 5.2: Given a decision e-frame $\langle\{\{C_{ik}\}_{m_i}\}_m, P_0, V_0\rangle$, the *expected value* $E(C_i)$ of a consequence set $\{C_{ik}\}_{m_i}$ is the sum $\sum_k p_{ik} \cdot v_{ik} = p_{i1} \cdot v_{i1} + p_{i2} \cdot v_{i2} + \ldots + p_{im_i} \cdot v_{im_i}$ over all consequences C_{ik} in the set.[1]

Example 5.1 (cont'd): Consider the same decision situation as above and the decision e-frame containing the same data. The expected value $E(C_1)$ is $0.35 \cdot 0.20 + 0.45 \cdot 0.55 + 0.20 \cdot 0.80 = 0.4775$. ■

The use of the principle of maximising the expected value (PMEV) dates several hundred years back, preceding the formal area of mathematical statistics and instead originating from pure monetary gambling. Over the years, a number of problems have been discovered with the principle. First, a well-known problem with PMEV is discussed, and thereafter, some alternative decision rules are reviewed.

A serious paradox was suggested by Allais [A53].[2] In this paradox, there is a game to be played and a reliable source of money that will guarantee that the game is carried through, regardless of its outcome.

[1] The definition is a slight abuse of notation, since the expectation operator should operate on a stochastic variable, but the stochastic variable represents exactly the corresponding consequence set.

[2] In this version, the actual numbers are adjusted for inflation since the 1950s.

Suppose that the following game is presented, perhaps being more like an offer. There are no stakes, i.e. no chance of losing money. There is a mandatory choice between the alternatives A and B, and all probabilities are fair in the sense that they are exactly as stated.

Alternative A: The player will receive $10 million for sure.

Alternative B: The player will have a 10% chance of receiving $50 million, an 89% chance of receiving $10 million, and a 1% chance of receiving nothing at all.

Next, another similar game is offered. There is a mandatory choice between C and D.

Alternative C: The player will have an 11% chance of receiving $10 million and an 89% chance of receiving nothing at all.

Alternative D: The player will have a 10% chance of receiving $50 million and a 90% chance of receiving nothing at all.

Many people tend to choose A over B and D over C. This violates the PMEV, no matter what utility values are assigned to the respective outcomes. In essence, the argument is that A and C are nearly the same, as are B and D, the difference being the first 11% of the probability mass, which differs in the same way for both of the pairs. See for example [S72] for a mathematical argument. Regardless of this fact, in experiments where it was subsequently pointed out to subjects who understood the mathematical argument, up to 1/3 retained their choice in spite of this.

Recently, Malmnäs suggested a way of avoiding the paradox by considering the choices as pairs (A, C), (A, D), (B, C), and (B, D) [M96]. Then consistent utility functions can be found that describe the choice of A over B and D over C. This requires, however, that the choice situations are considered in parallel, which is the case at least in the post-stage when the subjects are given an explanation on why they were being inconsistent.

Replacement Rules

Such problems with PMEV warrant further investigation, and several researchers, not least within economics, have proposed a number of alternative decision rules to replace (or sometimes supplement) the PMEV. Fishburn [F83] suggests an evaluation based on the quotient between two separate expected values, which has the following form

$$\frac{E(C_i, f_1)}{E(C_i, f_2)}$$

where f_1 and f_2 are two functions of the values involved.

Researchers such as Loomes and Sudgen [LS82] bring regret or disappointment into the evaluation to cover cases where numerically equal results are appreciated differently depending on what was once in someone's possession. Their suggested formula has the form

$$\sum_{k=1}^{n} p_{ik} \cdot (v_{ik} + R(v_{ik} - E(C_i)))$$

where R is supposed to be a regret function related to the ordinary expected value.

Some researchers, among them Quiggin [Q82], try to resolve the problems by requiring functions to modify the probabilities in the evaluation rule such as

$$\sum_{k=1}^{n} (f(s_{ik}) - f(s_{i(k-1)})) \cdot v_{ik}$$

where f is a strictly increasing function, the p_{ij}'s are in increasing v_{ij} order, and $s_{ik} = \sum_{j=1}^{k} p_{ij}$. Yaari [Y87] has pointed out that under certain reasonable assumptions, it must be the case that $f(p_{ij}) = p_{ij}$, and then he made the following extended suggestion

$$\sum_{k=1}^{n} (f(1 - s_{i(k-1)}) - f(1 - s_{ik})) \cdot v_{ik} + f(p_{im_i}) \cdot v_{im_i}$$

where s_{ij} is as above.

None of these suggestions are without problems. Malmnäs shows for those above and for some other proposals that their performance at best almost equals that of the expected value and at worst is much poorer, for example not even being consistent with first order stochastic dominance [M96]. All evaluation rules are subject to counter-examples similar to Allais'. Some of the simpler counter-examples that are problematic for many other rules are not so for the expected value. Still, the DELTA method allows for using numeric selection rules other than the expected value.

Supplementary Rules

There seem to be no compelling reasons to altogether reject the use of the PMEV, but since there exists no absolutely rational decision rule, a reasonable decision method should provide possibilities for evaluating decision situations in several respects. In many decision contexts, the decision maker may want to exclude particular alternative courses of action that are, in some way, too risky. This might be done by a class of supplementary decision rules called qualitative sorting or security levels. While an evaluation of a consequence set may result in an acceptable expected value, the consequences of selecting it might be so dire that it should nevertheless be avoided. It might, for example, endanger the entire purpose of the decision context, and in that case even a consequence with a low probability is too risky to neglect. In order to attain a high level of security and to be able to trust evaluations based on the information, it has been suggested that security levels should be imposed.

Such exclusions can be dealt with by specifying a security level for the probability and a threshold for the value. Then a consequence set would be undesirable if it violates both of these settings. Malmnäs' proposal is to supplement the expected value with qualitative evaluations, as was first suggested in [M94a]. An example is the qualitative sorting function, further developed by Ekenberg in [E94]. It has the following basic form

$$S(C_i, r, s) = \left(\sum_{v_{ij} \le r} p_{ij} \le s\right)$$

where r is the minimally tolerable value threshold and s is the maximally acceptable probability for events below the threshold to occur. This is a boolean function sorting out unwanted consequence sets. An example is given below under first order dominance, and an application of security levels can be found in Appendix A.

To sum up, the key observation is that there seems to be no perfect evaluation rule, although the expected value is found to be at least as good as many of its contenders. To improve that rule (or any other numeric rule), one way is to complement it with supplementary rules rather than engaging in further modifications of replacement rules in pursuit of the perfect rule.

DELTA Dominance

In this section, a general dominance rule is suggested as a unifying concept. In its generic form, it describes the type of dominance to be considered and thus the type and amount of computation involved in evaluating consequence sets in the framework. It can make use of many of the above suggested evaluation functions, even though the expected value is by far the most common. For convenience, a shorthand notation for the awkwardly long difference in expected values is introduced.

Definition 5.3: Given a decision e-frame $\langle \{\{C_{ik}\}_{m_i}\}_m, P_0, V_0 \rangle$, δ_{ij} denotes the expression $\sum_k p_{ik} \cdot v_{ik} - \sum_k p_{jk} \cdot v_{jk} =$ $p_{i1} \cdot v_{i1} + p_{i2} \cdot v_{i2} + \ldots + p_{im_i} \cdot v_{im_i} - p_{j1} \cdot v_{j1} - p_{j2} \cdot v_{j2} - \ldots - p_{jm_j} \cdot v_{jm_j}$ over all consequences in the consequence sets C_i and C_j.

In order to describe the dominance, a couple of concepts are required. The index set pair captures the consequences within each of the consequence sets that possess some desired property, in this case their value being at least as great as a given parameter.

Definition 5.4: Given a decision e-frame $\langle C,P_0,V_0\rangle$ and a real number $d \in [0,1]$, an *index set pair* $(K_i,K_j)(d)$ is $K_i = \{k$ ™ $v_{ik} \geq d\}$ and $K_j = \{k$ ™ $v_{jk} \geq d\}$.

When the parameter d varies over some range, the content of the index set may vary as well. The set of all such index sets is defined next.

Definition 5.5: Given a decision e-frame $\langle C,P_0,V_0\rangle$ and real numbers $a,b,d \in [0,1]$, $M_{ij}[a,b]$ is the set $\{(K_i,K_j)(d)$ ™ $d \in [a,b]\}$.

Those two definitions enable the following compact definition of the Δ-dominance, a key concept in this thesis from which the DELTA method takes its name. The idea behind the dominance is a pairwise comparison of the consequence sets employing the desired numerical function. The function is the same for both consequence sets. Note that the weak inequality must hold for all index set members, i.e. over the full interval range I, as specified.

Definition 5.6: Given a decision e-frame $\langle C,P_0,V_0\rangle$,
a function f, and two parameters $\alpha(P_0,V_0)$ and $\beta(P_0,V_0)$,
C_i *Δ[I]-dominates* C_j **iff**
$\forall\ (K_i,K_j)(d) \in M_{ij}[I]\ \sum_{k\in K_i} f(p_{ik}, v_{ik}, \alpha) - \sum_{k\in K_j} f(p_{jk}, v_{jk}, \beta) \geq 0$ and
$\exists\ (K_i,K_j)(d) \in M_{ij}[I]\ \sum_{k\in K_i} f(p_{ik}, v_{ik}, \alpha) - \sum_{k\in K_j} f(p_{jk}, v_{jk}, \beta) > 0$.[3]

This is a very general definition, and many instantiations are possible. In this thesis, a few are given and it is shown that some well-known evaluation concepts are special cases of Δ-dominance. The first subdivision of the Δ-dominance is into dominance orders depending on the function employed in the evaluation. The first and second orders are

[3] To simplify the presentation, the second condition is omitted in the sequel.

specifically addressed below, while the higher orders possible from the definition of Δ[I]-dominance are not further discussed.

First Order Dominance

The Δ-dominance is of the first order if the function used is a function of the probabilities only.

> **Definition 5.7:** Given a decision e-frame $\langle C,P_0,V_0\rangle$ and functions f and g, C_i *1[I]-dominates* C_j **iff** C_i Δ[I]-dominates C_j with $f(p_{ik},v_{ik},\alpha) = g(p_{ik})$ and $f(p_{jk},v_{jk},\beta) = g(p_{jk})$.

Thus, first order specialisation turns dominance into a difference of sums of a function of probabilities.

> **Note:** C_i *1[I]-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[I]$ $\sum_{k\in K_i} g(p_{ik}) \geq \sum_{k\in K_j} g(p_{jk})$.

The note shows some resemblance with a couple of familiar dominance concepts. One further specialisation of the first order Δ-dominance is the first order stochastic dominance, a well-known concept, not least in economics. To reach there, the general first order Δ-dominance is considered. It consists of specifying the range for the index set pairs to be the full [0,1] range.

> **Definition 5.8:** Given a decision e-frame $\langle C,P_0,V_0\rangle$, C_i *1S-dominates* C_j **iff** C_i 1[0,1]-dominates C_j.

When the function g employed is the simple $g(p_{ik}) = p_{ik}$ the general stochastic dominance turns into the commonly used first order stochastic dominance, which in the Δ-dominance concept is a specialisation of function as well as of index set range.

> **Definition 5.9:** Given a decision e-frame $\langle C,P_0,V_0\rangle$, C_i *1SE-dominates* C_j **iff** C_i 1S-dominates C_j with $g(p_{ik}) = p_{ik}$.

To see that this is indeed the ordinary first order stochastic dominance as claimed, it is convenient to make the following note, in which the form for 1SE-dominance coincides with the definition of first order stochastic dominance.

Note: C_i *1SE-dominates* C_j **iff** $\forall$ $(K_i,K_j)(d) \in M_{ij}[I]$ $\sum_{k \in K_i} p_{ik} \geq \sum_{k \in K_j} p_{jk}$.

Example 5.2: Consider a decision situation involving two consequence sets C_1 and C_2 that have three consequences each. The decision e-frame contains the following data.

$p_{11} = 0.35$ $\quad p_{21} = 0.30$
$p_{12} = 0.45$ $\quad p_{22} = 0.45$
$p_{13} = 0.20$ $\quad p_{23} = 0.25$
$v_{11} = 0.20$ $\quad v_{21} = 0.30$
$v_{12} = 0.55$ $\quad v_{22} = 0.70$
$v_{13} = 0.80$ $\quad v_{23} = 0.85$

For some calculation examples, take $(K_1,K_2)(0.4)$ where $\sum_k p_{1k} = 0.65$ and $\sum_k p_{2k} = 0.70$, or $(K_1,K_2)(0.6)$ where $\sum_k p_{1k} = 0.20$ and $\sum_k p_{2k} = 0.70$. In fact, for any legitimate index pair $(K_1,K_2)(d)$, $\sum_k p_{2k} - \sum_k p_{1k} \geq 0$, and thus C_2 1SE-dominates C_1. This can be seen in the graph in Figure 5.1, where the values are plotted against the cumulative mass function (cmf).[4] For C_2 to 1SE-dominate C_1, the curves may not cross, and the curve for C_2 must be below or on that of C_1 for all index set pairs. ■

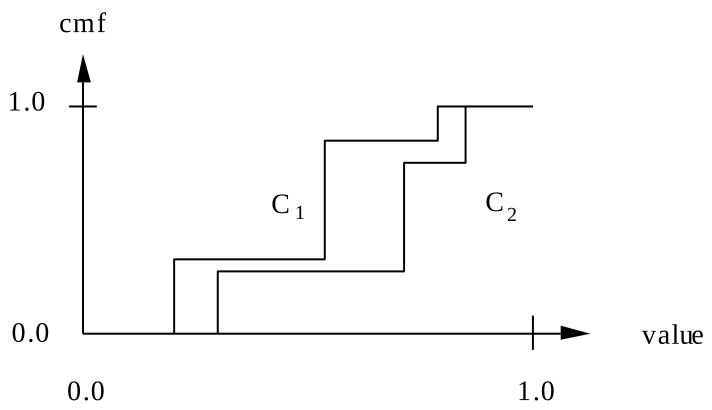

Figure 5.1 1SE-dominance

[4] Note that the probabilities p_{jk} in the cmf within a consequence set must be selected in increasing v_{jk} order for the graph to be meaningful.

Above, a supplementary function was mentioned under the name of qualitative sorting or security levels. This was a kind of threshold function separating wanted and unwanted outcomes (or desirable and undesirable consequence sets) according to a threshold rule applicable to the evaluation situation. This type of evaluation rule also turns out to be a special case of the Δ-dominance, viz. the dominance of a reference consequence set, i.e. the threshold.

> **Definition 5.10:** Given a decision e-frame $\langle C,P_0,V_0\rangle$ and two real numbers $s,t \in [0,1]$, C_j *violates general security level s for threshold value t* **iff** C_t $1[t,t]$-dominates C_j, where C_t is a consequence set with two consequences, $g(p_{t1}) = 1–g(s)$, $v_{t1} = 1$, $g(p_{t2}) = g(s)$, $v_{t2} = 0$.

When the function g is the simple $g(p_{ik}) = p_{ik}$, then the general security level turns into the ordinary security level concept, which again is a specialisation of both function and index set range.

> **Definition 5.11:** Given a decision e-frame $\langle C,P_0,V_0\rangle$ and two real numbers $s,t \in [0,1]$, C_j *violates security level s for threshold value t* **iff** C_j violates general security level s for threshold value t with $g(p_{jk}) = p_{jk}$.

To see that this is indeed the same concept as the security levels discussed above, the following note can be helpful. Note that there can only be one index set pair since the range of the value interval only contains r.

> **Note:** C_j *violates security level s for threshold value t* **iff** for $K_j = \{k ™ v_{jk} \geq t\}$ $\sum_{k\in K_j} p_{jk} \leq 1 - s$.

Example 5.3: Consider a decision situation involving two consequence sets C_1 and C_3 having three and four consequences respectively. The decision e-frame contains the following data.

$p_{11} = 0.35$ $\quad$ $p_{31} = 0.25$
$p_{12} = 0.45$ $\quad$ $p_{32} = 0.40$

$p_{13} = 0.20$	$p_{33} = 0.10$
	$p_{34} = 0.25$
$v_{11} = 0.20$	$v_{31} = 0.15$
$v_{12} = 0.55$	$v_{32} = 0.85$
$v_{13} = 0.80$	$v_{33} = 0.05$
	$v_{34} = 0.60$

The security level 5% for value 0.10 is violated by C_3 since consequence C_{33} has the value 0.05 (< 0.10) and occurs with a probability of 10% (> 5%), but not violated by C_1. ■

Second Order Dominance

The Δ-dominance is of the second order if the function used is a function of the probabilities and values only.

Definition 5.12: Given a decision e-frame $\langle C,P_0,V_0\rangle$ and functions f and h, C_i *2[I]-dominates* C_j **iff** C_i Δ[I]-dominates C_j with $f(p_{ik},v_{ik},\alpha) = h(p_{ik},v_{ik})$ and $f(p_{jk},v_{jk},\beta) = h(p_{jk},v_{jk})$.

Then the domination turns into a difference of sums of a function of probabilities and values.

Note: C_i *2[I]-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[I]$
$$\sum_{k\in K_i} h(p_{ik}, v_{ik}) \geq \sum_{k\in K_j} h(p_{jk}, v_{jk}).$$

As for the first order, a further specialisation into second-order stochastic dominance is possible. This is a well-known concept as well, and it turns out to be another case of Δ-dominance. First, the general second-order stochastic dominance is defined. As in the first order case, it consists of specifying the range for the index set pairs to be the full [0,1] range.

Definition 5.13: Given a decision e-frame $\langle C,P_0,V_0\rangle$, C_i *2S-dominates* C_j **iff** C_i 2[0,1]-dominates C_j.

If the function h employed is the most common $h(p_{ik},v_{ik}) = p_{ik}\cdot v_{ik}$, then the dominance turns into the commonly used second-order stochastic dominance, which in the Δ-dominance concept is a specialisation both of function and of index set range.

Definition 5.14: Given a decision e-frame $\langle C,P_0,V_0\rangle$, C_i *2SE-dominates* C_j **iff** C_i 2S-dominates C_j with $h(p_{ik},v_{ik}) = p_{ik}\cdot v_{ik}$.

To see explicitly that the definition sequence has arrived at the ordinary second-order stochastic dominance, it is helpful to make the following note, in which the form for 2SE-dominance can be seen to be almost equivalent to the textbook definition of second-order stochastic dominance.[5]

Note: C_i *2SE-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[0,1]$ $\sum_{k\in K_i} p_{ik}\cdot v_{ik} \geq \sum_{k\in K_j} p_{jk}\cdot v_{jk}$.

Example 5.2 (cont'd): The decision situation is augmented by a fourth consequence set C_4, also having three consequences, to be compared with C_1. The decision e-frame contains the following data.

$p_{11} = 0.35$ $\quad p_{41} = 0.40$
$p_{12} = 0.45$ $\quad p_{42} = 0.40$
$p_{13} = 0.20$ $\quad p_{43} = 0.20$
$v_{11} = 0.20$ $\quad v_{41} = 0.40$
$v_{12} = 0.55$ $\quad v_{42} = 0.65$
$v_{13} = 0.70$ $\quad v_{43} = 0.80$

For some calculation examples, take $(K_1,K_4)(0.5)$ where $\sum_k p_{1k} = 0.65$ and $\sum_k p_{4k} = 0.60$, or $(K_1,K_4)(0.6)$ where $\sum_k p_{1k} = 0.20$ and $\sum_k p_{4k} = 0.60$. This time, for some legitimate index pairs $(K_1,K_4)(d)$ $\sum_k p_{4k} - \sum_k p_{1k} \geq 0$, and for others $\sum_k p_{1k} - \sum_k p_{4k} \geq 0$. Thus C_4 does not 1SE-dominate C_1 or vice versa. For the same example values, $(K_1,K_4)(0.5)$ yields $\sum_k p_{1k}\cdot v_{1k} = 0.3875$ and $\sum_k p_{4k}\cdot v_{4k} = 0.42$, and $(K_1,K_4)(0.6)$ yields $\sum_k p_{1k}\cdot v_{1k} = 0.14$ and $\sum_k p_{4k}\cdot v_{4k} = 0.42$. Now, for all legitimate index pairs $(K_1,K_4)(d)$ $\sum_k p_{4k}\cdot v_{4k} -$

[5] This is a slight simplification. Since p and v are multiplied, both sums should run from the same d-value in $(K_i,K_j)(d)$. For each index set pair, there might be one compensation term in one of the two sums.

$\sum_k p_{1k} \cdot v_{1k} \geq 0$, and the conclusion is that C_4 2SE-dominates C_1. This is illustrated in Figure 5.2, where the values are plotted against the cumulative mass function (cmf). For C_4 to 2SE-dominate C_1, the curves may cross, but the area under the curve for C_4 must be less or equal to that of C_1 for all index set pairs. ■

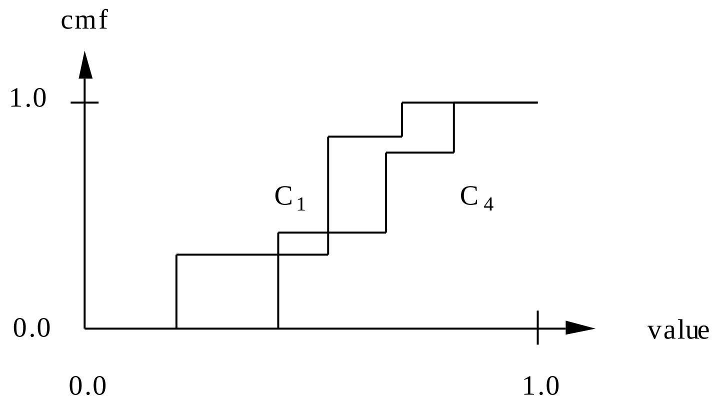

Figure 5.2 2SE-dominance

Another second order Δ-dominance is the ordinary expected value and some of the suggested replacements.[6] One of their characteristics is that they evaluate only by full index set pairs, i.e. pairs that contain all members of each consequence set. The general numerical dominance is a straightforward specialisation of 2Δ-dominance.

Definition 5.15: Given a decision e-frame $\langle C,P_0,V_0 \rangle$, C_i *N-dominates* C_j **iff** C_i 2[0,0]-dominates C_j.

This corresponds to the evaluation rules that apply a probability and value formula to the consequence set in order to reach a numerical verdict on which one is preferable. The last specialisation of the second order is the ordinary expected value, which is termed NE-dominance and is realised by letting $f(p_{ik},v_{ik}) = p_{ik} \cdot v_{ik}$ in the N-dominance.

[6] Other replacements that, for example, use the sum of ordered probabilities, are categorised as higher order, but reasoning similar to N-dominance is applicable to them as well.

Definition 5.16: Given a decision e-frame $\langle C,P_0,V_0\rangle$, C_i *NE-dominates* C_j **iff** C_i N-dominates C_j with $h(p_{ik},v_{ik}) = p_{ik}\cdot v_{ik}$.

This can be seen to be the expected value, since the only index set pair generated by the [0,0]-range is the pair of complete consequence sets.

Note: C_i *NE-dominates* C_j **iff** for $(K_i,K_j)(0)$ $\delta_{ij} \geq 0$.[7]

Also, note that $\delta_{ij} \geq 0$ is not applicable to 2SE-dominance since it involves different index set pairs while NE-dominance always applies only to the full index sets of the consequence sets.

Example 5.3 (cont'd): Consider again the decision situation involving the consequence sets C_1 and C_3. To recapitulate, the decision e-frame contains the following data.

$p_{11} = 0.35$	$p_{31} = 0.25$
$p_{12} = 0.45$	$p_{32} = 0.40$
$p_{13} = 0.20$	$p_{33} = 0.10$
	$p_{34} = 0.25$
$v_{11} = 0.20$	$v_{31} = 0.15$
$v_{12} = 0.55$	$v_{32} = 0.85$
$v_{13} = 0.80$	$v_{33} = 0.05$
	$v_{34} = 0.60$

C_3 NE-dominates C_1 since $E(C_3) = 0.5325$ and $E(C_1) = 0.4775$. Above, the security level of 5% for value 0.10 was violated by C_3 but not by C_1. Thus, the two rules may recommend different consequence sets. ■

Any particular implementation of the Δ-dominance will use a selection of dominance rules as appropriate. That is a prime reason for introducing a sequence of them here. For example, if the expected value is preferred to other numerical rules, the selection of 1SE-, 2SE-, and NE-dominance is a plausible one.

[7] Actually $\delta_{ij} > 0$ if the complete definition of Δ-dominance is considered.

GAMMA Dominance

Sometimes, keeping the computational load to a minimum is of great importance, even at the expense of obtaining exact results. The Δ-dominance makes pairwise comparisons between the consequence sets, leading to $m \cdot (m-1)/2$ computations for m consequence sets. To reduce the number of comparisons to m, the Γ-dominance is introduced. The idea of Γ-dominance is to compare each consequence set to all others (or a subset thereof) at the same time by forming a weighted average of the remaining ones and studying their difference.

> **Terminology:** Given a decision e-frame $\langle \{C_i\}_m, P_0, V_0 \rangle$, the *polar index set* for C_i is $J = \{1,\ldots,m\} \setminus \{i\}$.

In general, the indices in such a set could be any subset of the indices in the frame excluding i. According to the terminology clause, in this thesis J will always be all other indices (all other consequence sets) in the frame.

> **Definition 5.17:** Given a decision e-frame $\langle C, P_0, V_0 \rangle$ and a real number $d \in [0,1]$, an *index set tuple* $(K_i, \{K_j\}_{j \in J})(d)$ is $K_i = \{k$ ™ $v_{ik} \geq d\}$ and $K_j = \{k$ ™ $v_{jk} \geq d\}$, $\forall\, j \in J$.

Similar to the Δ definitions, the set of all index set tuples is useful.

> **Definition 5.18:** Given a decision e-frame $\langle C, P_0, V_0 \rangle$ and real numbers $a,b,d \in [0,1]$, $M_i[a,b]$ is the set $\{(K_i, \{K_j\}_{j \in J})(d)$ ™ $d \in [a,b]\}$.

The Γ-dominance can now be introduced. It can be thought of as a "setwise" comparison of one consequence set to many others, employing the desired numerical function. It is a straightforward generalisation of Δ-dominance, and for decision frames with only two consequence sets they coincide. The function f must be the same for both the consequence set and the set of sets. The weak inequality must hold for all index set members, i.e. over the full interval range I.

Definition 5.19: Given a decision e-frame $\langle C,P_0,V_0\rangle$, a function f, and two parameters $\alpha(P_0,V_0)$ and $\beta(P_0,V_0)$, C_i *Γ[I]-dominates* $\{C_j\}_{j\in J}$ **iff** $\forall$ $(K_i,\{K_j\}_{j\in J}) \in M_i[I]$

$$\sum_{k\in K_i} f(p_{ik}, v_{ik}, \alpha) - \frac{1}{n-1}\sum_{j\in J}\left(\sum_{k\in K_j} f(p_{jk}, v_{jk}, \beta)\right) \geq 0.$$

From the definition of Γ-dominance, the same set of evaluation rules as for Δ-dominance can be defined. The insights gained from explicitly stating them in this thesis are minimal. Suffice it to mention that all the resulting rules behave as expected and that for a problem with only two consequence sets all Γ definitions coincide with their Δ counterparts.

Γ-dominance is an approximate concept, but still leads to the same ranking as its Δ counterpart. To realise this, consider a decision situation with three alternatives A_1, A_2, and A_3 modelled as consequence sets C_1, C_2, and C_3 having expected values $E(C_1)$, $E(C_2)$, and $E(C_3)$ respectively. Assume without loss of generality that $E(C_1) > E(C_2) > E(C_3)$. Then an evaluation results in $\delta_{12} > 0$ and $\delta_{23} > 0$. Using the Γ-version of NE-dominance, use differences $\gamma_i = E(C_i) - \frac{1}{n-1} \cdot \sum_k E(C_k)$. Now $\gamma_1 - \gamma_2 = E(C_1) - 0.5\cdot(E(C_2)+E(C_3)) - E(C_2) + 0.5\cdot(E(C_1)+E(C_3)) = 1.5\cdot(E(C_1) - E(C_2)) > 0$ by the assumption. Likewise, $\gamma_2 - \gamma_3 = E(C_2) - 0.5\cdot(E(C_1)+E(C_3)) - E(C_3) + 0.5\cdot(E(C_1)+E(C_2)) = 1.5\cdot(E(C_2) - E(C_3)) > 0$. Finally $\gamma_1 - \gamma_3 > 0$ by transitivity and thus the ranking is preserved. It can be generalised to any number of consequence sets and also to other rules.

This concludes the second section of the evaluation chapter. The last section deals with evaluation of decision frames with imprecise information.

Frame Evaluation

In the rest of the chapter, ordinary $\langle C,P,V\rangle$ decision frames with all kinds of constraints are again considered. Various lines of thought have emerged in response to the problem arising when the information given is imprecise and overlaps in the sense that parts of the information seem to favour one alternative (consequence set) while other parts favour another one.

The first idea might be to try and develop concepts based on ${}^{PV}\min(N(C_i))$ and ${}^{PV}\max(N(C_i))$. Those are absolute values and delimit the values that $N(C_i)$ may assume. They provide an overview of the decision situation, but in most interesting cases, the ranges overlap for different consequence sets. The idea could then be to compare the differences in minima and maxima of the consequence sets respectively.

Definition 5.20: Given a decision frame $\langle C,P,V \rangle$, C_i is *Ψ-better than* C_j **iff** ${}^{PV}\min(N(C_i)) - {}^{PV}\min(N(C_j)) > 0$ and ${}^{PV}\max(N(C_i)) - {}^{PV}\max(N(C_j)) > 0$.

Now, the consequence sets can be ranked in a partial order according to Ψ-better than, but only in some cases will the order be complete. In other cases, some pair of consequence sets C_i and C_j will have ${}^{PV}\min(N(C_i)) > {}^{PV}\min(N(C_j))$ and ${}^{PV}\max(N(C_i)) < {}^{PV}\max(N(C_j))$ or vice versa. Which boundary should then take precedence? Ambiguities like this have led to other approaches, in which comparing differences for consequence sets and using concepts of dominance are important elements.

Admissibility

The first attempt to compare the consequence sets using differences is fetched from statistical decision theory. There, the decision rules are based on the expected value $E(C_i)$. Within statistical decision theory, the following definitions, adapted from [L59], are common.[8]

Definition 5.21: Given a decision frame $\langle C,P,V \rangle$, C_i is *at least as good* as C_j **iff** $\delta_{ij} < 0$ is inconsistent with $P \cup V$.

[8] Since the definitions originate from statistics they are based on the expected value, but the reasoning can easily be applied to other numerical rules as well.

C_i is *better* than C_j **iff** C_i is at least as good as C_j and
$\delta_{ij} > 0$ is consistent with $P \cup V$.

If there are more than two alternatives, some criterion is needed by which to compare an alternative to any number of other alternatives at the same time.

Definition 5.22: C_i is *admissible* **iff** no other C_j is better.

However, the following observation much clearer shows the computational meaning of admissibility.

Observation: Given a decision frame $\langle C,P,V \rangle$,
C_i is *admissible* **iff** for each $j \in J$:
(i) $\{\delta_{ij} > 0\} \cup P \cup V$ is consistent, or
(ii) $\{\delta_{ji} > 0\} \cup P \cup V$ is inconsistent.

Proof: According to Definition 5.22 C_i is admissible **iff** no other C_j is better. For a specific C_j it is true that
C_j is better than C_i **iff**
($\delta_{ij} > 0$ inconsistent with $P \cup V$) $\wedge$
($\delta_{ji} > 0$ consistent with $P \cup V$).
Thus the negation "C_j is *not* better than C_i" can be expressed as
$\neg$[($\delta_{ij} > 0$ inconsistent with $P \cup V$) $\wedge$
($\delta_{ji} > 0$ consistent with $P \cup V$)].
The negation expands into the disjunction
$\neg$($\delta_{ij} > 0$ inconsistent with $P \cup V$) $\vee$
$\neg$($\delta_{ji} > 0$ consistent with $P \cup V$).
This can finally be written as
($\delta_{ij} > 0$ consistent with $P \cup V$) $\vee$
($\delta_{ji} > 0$ inconsistent with $P \cup V$).
All derivation steps are equivalencies and valid for any C_j, $j \neq i$, in the frame. Thus both directions of the **iff** are proven. ■

This shows why it is necessary to take two different clauses into consideration when determining admissibility. Evaluating the consequence sets in a decision frame, one of two situations may occur. Either

(i) only one consequence set is admissible, or

(ii) more than one consequence set is admissible.

For case (i) the task is done, since the only remaining consequence set is superior to all the others, i.e. it dominates them all. Case (ii) remains, which is the usual and interesting case. There is some overlap in the values that the expected value can take on for the different consequence sets. It is not at all obvious how it should be handled. Different authors have suggested various solutions. Levi, for example, considers situations where both probabilities and values are represented by intervals bounded by upper and lower limits [L74]. He then defines a hierarchy of admissibility concepts still building on the idea of any single instance being superior for a particular consequence set. Additional rules include preservation of options, i.e. a consequence set is better if more optional future actions are possible (cf. options theory and contingent claims analysis [H89]), and less spread or less risk, i.e. a kind of security level. Malmnäs criticises Levi and other researchers in [M94a] and proposes the introduction of a slack parameter (t) into the admissibility concept, yielding the t-admissibility. Ekenberg further develops it by employing the concept of proportion to measure in how large parts of the bases different values of t hold for the t-admissibility of the respective consequence sets [E94], although that leads to some contradictions when such a procedure is applied, which is shown in the required graduation position paper preceding this thesis [D97b].

Strength Concepts

This thesis takes another approach to the problem of evaluating interval decision problems. The *strength* of a consequence set C_i compared to another set C_j, given as a number ${}^{PV}\max(\Delta_{ij}) \in [-1,1]$, shows how the most favourable consistent assignments of numbers to the probability and value variables lead to the largest difference between the consequence sets.

Terminology: Given a decision frame $\langle C,P,V \rangle$ and an index set pair $(K_i,K_j)(d)$, Δ_{ij} denotes an instance of $\sum_{k \in K_i} f(p_{ik}, v_{ik}, \alpha) - \sum_{k \in K_j} f(p_{jk}, v_{jk}, \beta)$.

To begin with, three strength concepts are introduced, on which the evaluation principles will be based.

Definition 5.23: Given a decision frame $\langle C,P,V \rangle$, the *maximal difference* Δ_{ij} in the frame is ${}^{PV}\max(\Delta_{ij})$ and the *minimal difference* Δ_{ij} is ${}^{PV}\min(\Delta_{ij})$.

Thus, the maximal and minimal differences are in a sense the most and least favourable possibilities respectively. They are both extreme results as they require every single probability and value variable to take on its most or least desirable numerical value at the same time. Moreover, there is an element of comparison inherent in a decision procedure. The evaluation results are interesting in comparison to the results of the other consequence sets. Hence, it is reasonable to consider the differences in strength as well. Then it makes sense to evaluate the relative strength of C_i compared to C_j in addition to the strengths themselves, since such strength values are compared to some other strengths anyway in order to rank the consequence sets. To accomplish this, the medium difference is introduced.

Definition 5.24: Given a decision frame $\langle C,P,V \rangle$, let $\alpha \in [0,1]$ be real number. The *α-medium difference* Δ_{ij} in the frame is ${}^{PV}[\alpha]\mathrm{mid}(\Delta_{ij}) = \alpha \cdot {}^{PV}\max(\Delta_{ij}) + (1-\alpha) \cdot {}^{PV}\min(\Delta_{ij})$. The *medium difference* Δ_{ij} in the frame is ${}^{PV}\mathrm{mid}(\Delta_{ij}) = {}^{PV}[0.5]\mathrm{mid}(\Delta_{ij})$.

α can be considered a precedence parameter that indicates if one boundary should be given more weight than the other. The medium is also the relative strength as discussed informally in Chapter 2, i.e. the difference in maximal Δ-values when the frame is considered from the viewpoint of each consequence set respectively. Thus, it is a measure of difference in strength between the consequence sets.[9] This duality view is a key to understanding the selection process proposed later.

Note: The *relative strength* of C_i compared to C_j in a decision frame is $^{PV}\text{mid}(\Delta_{ij}) = \frac{^{PV}\max(\Delta_{ij}) - {}^{PV}\max(\Delta_{ji})}{2}$

For the expected value difference δ_{ij}, the concept of strength is related to statistical decisions in the following way.

Note: C_i is *at least as good* as C_j **iff** $^{PV}\min(\delta_{ij}) \geq 0$. C_i is *better* than C_j **iff** C_i is at least as good as C_j and $^{PV}\max(\delta_{ij}) > 0$. C_i is *admissible* **iff** no other C_j is better.

Strong, Marked, and Weak Dominance

The selection procedure suggested in this thesis is based on the expansion and contraction principles as introduced in Chapter 4 and on the concepts of strong, marked, and weak dominance as introduced below.

Dominance means that one consequence set is superior to another, at least in a part of the solution space to the bases. The weakest relation would be if "a part" refers to a single solution vector. A more reasonable interpretation of "a part" is if it is superior in a substantial fraction of the solutions. Dominance in the strongest sense would mean requiring that the "part" consists of all solution vectors. This idea is captured in the concepts of strong, marked, and weak dominances.[10] They correspond to the minimal, medium, and maximal differences.

[9] The definitions of $^{PV}\max(\Gamma_j)$, $^{PV}\min(\Gamma_j)$, and $^{PV}[\alpha]\text{mid}(\Gamma_j)$ are similar.

[10] For P_3 or V_3 bases and some Δ-dominance instances, $M_{ij}[I]$ might not be unique, but rather form a set. In such cases, all members of the set are evaluated and the minimal (maximal) result is used. For NE-dominance, the most important case, this cannot occur.

Definition 5.25: Given a decision frame $\langle C,P,V \rangle$,
C_i *strongly* Δ*[I]-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[I]$ ${}^{PV}\min(\Delta_{ij}) \geq 0$.
C_i *markedly* Δ*[I]-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[I]$ ${}^{PV}\text{mid}(\Delta_{ij}) \geq 0$.
C_i *weakly* Δ*[I]-dominates* C_j **iff** $\forall\ (K_i,K_j)(d) \in M_{ij}[I]$ ${}^{PV}\max(\Delta_{ij}) \geq 0$.

The selection procedure might proceed as follows. First, the various first order rules that are included in the particular procedure are applied in turn, for example (and most commonly) first order stochastic dominance and security levels. Possibly some consequence sets are then filtered out from the decision process. Next, the more general of the second-order rules are applied in the same manner. In the end, often the NE-domination remains, and usually a number of consequence sets are still being considered. For NE-dominance, the computational patterns are as follows.

Note: For the expected value rule,
C_i *strongly NE-dominates* C_j **iff** for $(K_i,K_j)(0)$

$${}^{PV}\min\left(\sum_{k\in K_i} p_{ik}\cdot v_{ik} - \sum_{k\in K_j} p_{jk}\cdot v_{jk}\right) \geq 0.$$

and similarly for marked and weak NE-dominance.

An example shows the use of NE-dominance.

Example 5.4: The decision involves three consequence sets C_1, C_2 and C_3. The sets C_2 and C_3 have one consequence each while C_1 has two. The corresponding decision frame contains the following statements:

$p_{11} \in [0.00, 1.00]$ $\quad v_{11} = 1.00$
$p_{12} \in [0.00, 1.00]$ $\quad v_{12} = 0.00$
$p_{11} + p_{12} = 1.00$
$p_{21} = 1.00$ $\quad v_{21} = 0.89$
$p_{31} = 1.00$ $\quad v_{31} = 0.88$

Since the example – for the sake of hand computability – involves mostly equality constraints, the calculations can be simplified by first considering the possible ranges for the δ_{ij}'s.

$\delta_{12} \in [-0.89, 0.11]$
$\delta_{13} \in [-0.88, 0.12]$
$\delta_{23} \in [0.01, 0.01]$

From the δ_{ij} ranges, the max, min, and mid are found to be as follows.

$^{PV}\max(\delta_{12}) = 0.11$	$^{PV}\min(\delta_{12}) = -0.89$
$^{PV}\max(\delta_{13}) = 0.12$	$^{PV}\min(\delta_{13}) = -0.88$
$^{PV}\max(\delta_{21}) = 0.89$	$^{PV}\min(\delta_{21}) = -0.11$
$^{PV}\max(\delta_{23}) = 0.01$	$^{PV}\min(\delta_{23}) = 0.01$
$^{PV}\max(\delta_{31}) = 0.88$	$^{PV}\min(\delta_{31}) = -0.12$
$^{PV}\max(\delta_{32}) = -0.01$	$^{PV}\min(\delta_{32}) = -0.01$

$^{PV}\text{mid}(\delta_{12}) = 0.5\cdot(0.11 - 0.89) = -0.39$
$^{PV}\text{mid}(\delta_{13}) = 0.5\cdot(0.12 - 0.88) = -0.38$
$^{PV}\text{mid}(\delta_{21}) = 0.5\cdot(0.89 - 0.11) = 0.39$
$^{PV}\text{mid}(\delta_{23}) = 0.5\cdot(0.01 - (-0.01)) = 0.01$
$^{PV}\text{mid}(\delta_{31}) = 0.5\cdot(0.88 - 0.12) = 0.38$
$^{PV}\text{mid}(\delta_{32}) = 0.5\cdot(-0.01 - 0.01) = -0.01$ ■

Expansion and Contraction

The expansion and contraction are generalised sensitivity analyses to be carried out in a large number of dimensions. In non-trivial decision situations, when a decision frame contains numerically imprecise information, the different principles suggested above are often too weak to yield a conclusive result by themselves. Thus, after the elimination of undesirable consequence sets, the decision maker could still find that no conclusive decision has been made. One way to proceed could be to determine the stability of the relation between the consequence sets under consideration. A natural way to investigate this is to consider values near the boundaries of the constraint intervals as being less reliable than the core due to the former being deliberately imprecise. This is taken into account by measuring the dominated regions indirectly using the concepts of expansion and contraction.

The principles can be motivated by the difficulties of performing simultaneous sensitivity analysis in several dimensions at the same time. It can be hard to gain a real understanding of the solutions to large decision problems using only low-dimensional analyses, since different combinations of dimensions can be critical to the evaluation results. Investigating all possible such combinations would lead to a procedure of high combinatorial complexity in the number of cases to investigate. Using expansions (and contractions), such difficulties are circumvented. The idea behind the expansion principle is to investigate how much the core can be expanded before dominance disappears between the consequence sets compared. If there is no dominance in the original core, it may be contracted towards the focal point in order to achieve dominance. The expansion and contraction avoid the complexity inherent in combinatorial analyses, but it is still possible to study the stability of a result by gaining a better understanding of how important the constraint boundaries are. By co-varying the contractions of an arbitrary set of intervals, it is possible to gain much better insight into the influence of the structure of the decision frame on the solutions.[11] Contrary to volume estimates, expansions (and contractions) are not measures of the sizes of solution sets but rather of the strength of statements when the original solution sets are modified in controlled ways. Both the set of intervals under investigation and the scale of individual contractions can be controlled. Consequently, an expansion can be regarded as a focus parameter that zooms out from central sub-intervals (the core) to the full statement intervals. The selection procedure could then continue with:

[11] For a 100% contraction, the volume of each base is reduced to a single point. For this special case, the results coincide with the ordinary expected value.

(i) Remove all strongly NE-dominated consequence sets

(ii) If more than one consequence set remains

(ii a) Contract the frame until only one consequence set remains

(ii b) Remove the markedly NE-dominated consequence sets

(ii c) A combination of (ii a) and (ii b)

(iii) If only one consequence set remains

(iii a) Expand the frame until other consequence sets appear

(iii b) Study the markedly NE-dominated consequence sets

(iii c) A combination of (iii a) and (iii b)

This is not a very precise selection procedure, and it is not meant to be. Its particular instantiation depends on the decision situation, whether the decision maker is a human or a machine, and whether the goal is to make an ultimate decision or (very common for humans) to gain a better understanding of the decision problem.

For simplicity of presentation, the text and the examples in this chapter do not involve the concepts of core or expansion. Rather, the hull is contracted to the focal point, and in a sense, the core can be considered to coincide with the hull for those examples. Otherwise, the ideas to be pointed out with the examples might be lost in the calculations. Also for presentation reasons, the examples are small and contrived with unusually sized intervals. Some examples facilitate hand calculations to convey some idea, while others are machine-generated. Since no core is specified, the contraction goes from the hull inwards to the degree of 100%. The following three examples are from sample runs of the DDT text interface.

Example 5.5: Consider a decision situation involving two consequence sets C_1 and C_2 that have three consequences each. According to the DDT tool, the decision frame contains the following data.

```
Frame 'ex55' in folder 'PhD' has 2 alternatives
A1 (no_name1) with 3 consequences
A2 (no_name2) with 3 consequences

The probability base contains 6 constraints
```

```
 1: P1.1 ∈ [0.100,0.600]
 2: P1.2 ∈ [0.200,0.400]
 3: P1.3 ∈ [0.300,0.400]
 4: P2.1 ∈ [0.400,0.600]
 5: P2.2 ∈ [0.250,0.400]
 6: P2.3 ∈ [0.200,0.300]

Probability hull      Symmetry hull
P1.1 ∈ [0.200,0.500]  [0.200,0.500]
P1.2 ∈ [0.200,0.400]  [0.200,0.400]
P1.3 ∈ [0.300,0.400]  [0.300,0.400]
P2.1 ∈ [0.400,0.550]  [0.400,0.513]
P2.2 ∈ [0.250,0.400]  [0.250,0.363]
P2.3 ∈ [0.200,0.300]  [0.200,0.275]

The value base contains 6 constraints
 1: V1.1 ∈ [0.860,0.880]
 2: V1.2 ∈ [0.470,0.520]
 3: V1.3 ∈ [0.040,0.100]
 4: V2.1 ∈ [0.660,0.680]
 5: V2.2 ∈ [0.570,0.620]
 6: V2.3 ∈ [0.410,0.450]

Value hull
V1.1 ∈ [0.860,0.880]
V1.2 ∈ [0.470,0.520]
V1.3 ∈ [0.040,0.100]
V2.1 ∈ [0.660,0.680]
V2.2 ∈ [0.570,0.620]
V2.3 ∈ [0.410,0.450]

Focal point
Cons.    P      V
C1.1:  0.350  0.870
C1.2:  0.300  0.495
C1.3:  0.350  0.070
C2.1:  0.456  0.670
C2.2:  0.306  0.595
C2.3:  0.238  0.430

Contraction       0%     20%     40%     60%     80%    100%
E1 - E2 min: -0.241  -0.215  -0.189  -0.163  -0.138  -0.113
        mid: -0.113  -0.113  -0.113  -0.113  -0.113  -0.113
        max:  0.012  -0.013  -0.037  -0.062  -0.087  -0.113
```

The decision frame is of type P_1 and V_1, thus containing only range constraints. The evaluation reveals that consequence set C_2 is to prefer in almost all of the frame, even when hardly any contraction is applied. C_2 NE-dominates C_1 strongly from about 10% contraction and markedly for all contractions, a very stable result. ■

Example 5.6: Consider a decision situation involving two consequence sets C_1 and C_2 that have one consequence each. According to the DDT tool, the decision frame contains the following data.

```
Frame 'ex56' in folder 'PhD' has 2 alternatives
A1 (no_name1) with 1 consequence
A2 (no_name2) with 1 consequence

The probability base contains 0 constraints

Probability hull      Symmetry hull
P1.1 ∈ [1.000,1.000]  [1.000,1.000]
P2.1 ∈ [1.000,1.000]  [1.000,1.000]

The value base contains 3 constraints
 1: V1.1 - V2.1 ∈ [-0.100,0.100]
 2: V1.1 ∈ [0.400,0.800]
 3: V2.1 ∈ [0.100,0.500]

Value hull
V1.1 ∈ [0.400,0.600]
V2.1 ∈ [0.300,0.500]

Focal point
Cons.    P      V
C1.1:  1.000  0.500
C2.1:  1.000  0.400

Contraction       0%     20%     40%     60%     80%    100%
E1 - E2 min: -0.100  -0.060  -0.020   0.020   0.060   0.100
        mid:  0.000   0.020   0.040   0.060   0.080   0.100
        max:  0.100   0.100   0.100   0.100   0.100   0.100
```

The decision frame is of type P_1 and V_2, thus containing only range constraints in the probability base but also comparative constraints in the value base. This time, the evaluation shows that in the uncontracted frame, the consequence sets seem to be equal, but under contraction, C_1 is to prefer more the further the contraction continues. C_1 never NE-dominates C_2 strongly but dominates markedly for all contractions beyond 0%. This indicates that contraction is an essential component of the analysis. ■

Example 5.7: Consider almost the same decision situation as in Example 5.6. According to the DDT tool, the decision frame contains the following data.

```
Frame 'ex57' in folder 'PhD' has 2 alternatives
A1 (no_name1) with 1 consequence
A2 (no_name2) with 1 consequence

The probability base contains 0 constraints

Probability hull      Symmetry hull
P1.1 ∈ [1.000,1.000]  [1.000,1.000]
P2.1 ∈ [1.000,1.000]  [1.000,1.000]
```

```
The value base contains 3 constraints
 1: V1.1 - V2.1 ∈ [-0.100,0.050]
 2: V1.1 ∈ [0.400,0.800]
 3: V2.1 ∈ [0.100,0.500]

Value hull
V1.1 ∈ [0.400,0.550]
V2.1 ∈ [0.350,0.500]

Focal point
Cons.     P       V
C1.1:  1.000  0.475
C2.1:  1.000  0.425

Contraction       0%     20%     40%     60%     80%    100%
E1 - E2 min: -0.100  -0.070  -0.040  -0.010   0.020   0.050
        mid: -0.025  -0.010   0.005   0.020   0.035   0.050
        max:  0.050   0.050   0.050   0.050   0.050   0.050
```

This example illustrates that dominance may shift under contraction. The evaluation shows that in the uncontracted frame, A_2 is to prefer, but under contraction A_1 becomes stronger the longer the contraction continues. Beyond about 67% contraction, A_1 NE-dominates A_2 strongly but is itself dominated markedly for small contractions less than 33%. ■

This concludes the frame evaluation chapter, and thus the more formal evaluation aspects, at least from a definition point of view. The next chapter deals with trying to compute the numerical values of some of those definitions in order to turn it into a truly computational method.

All the roads jam up with credit
And there's nothing you can do
It's all just bits of paper
Flying away from you

Look out, world
Take a good look
What comes down here
You must learn this lesson fast
And learn it well
This ain't no upwardly mobile freeway
This is the road to hell

– C. Rea

6

Optimisation

To make a decision analysis method computational, and thus making it a method for real-life decisions, two main ingredients are necessary. The first is a suitable representation and evaluation rules of the decision problems, such as the method presented in Chapters 4–5. The other is reasonably fast computational algorithms, which is the topic of this chapter. Most of the demanding computations required by DELTA are optimisation-related algorithms.

The chapter is divided into three main sections. The first deals with calculating properties of decision frames using linear programming methods and the second deals with algorithms for computing evaluation rules by employing bilinear optimisation. The last section contains a discussion of the Simplex method and its implementation in the DELTA solver.

Frame Properties

In order to assess the properties of a frame, computational methods are required that can determine whether a given base has a particular property or not. One of the most fundamental components is a way of determining consistency in a base. Since the base consists of a linear system of inequalities, a natural candidate area for an algorithm is linear programming.

The area of linear programming (LP) was formed in the 1940s and has been a large and lively area of research ever since.[1] It deals with the maximising (or minimising) of a linear function with a large number of likewise linear constraints in the form of weak inequalities. Research efforts in the field are partly focused on developing clever algorithms for fast numerical computations. This chapter assumes that the reader is familiar with the basics of LP in general and with the Simplex method in particular. Those unfamiliar with these subjects may refer to any standard textbook on the subject, e.g. [BHM77, C83]. The LP problem is the following optimising problem:

max $f(\mathbf{x})$
when $\mathbf{Ax} \geq \mathbf{b}$
and $\mathbf{x} \geq \mathbf{0}$

where $f(\mathbf{x})$ is a linear expression of the type $k_1x_1 + k_2x_2 + \ldots + k_nx_n$, $\mathbf{Ax} \geq \mathbf{b}$ is a matrix inequality with rows $a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \geq b_1$ through $a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \geq b_m$, and $\mathbf{x} \geq \mathbf{0}$ are the non-negativity constraints $x_i \geq 0$ for each variable. Amongst all feasible points, the solution to $f(\mathbf{x})$ is sought that has the highest numerical value, i.e. the best solution vector $\mathbf{x}$, the components of which are all non-negative and satisfy all constraints. A minimum can be searched for by negating $f(\mathbf{x})$.

Consistency

The first algorithm is a procedure for determining whether a base is consistent or not. A base is consistent if any solution whatsoever can be found to the set of interval constraints. Note the similarities with the LP problem formulation. Let there be m interval constraints in the base. By

[1] Even though the ideas were around earlier, Danzig's timing was better. Mathematicians in the former Soviet Union formulated similar ideas already in the late 1930s (even including rudimentary algorithms) but with no computers available, their work was neglected, even domestically [GT89].

introducing new variables $y_1,\ldots,y_k$, with $k = 2\cdot m$, to the consistency problem, it can be reformulated as

min $(y_1 + \ldots + y_k)$
when $\mathbf{Ax} \geq \mathbf{b}$
and $\mathbf{x} \geq \mathbf{0}$, $\mathbf{y} \geq \mathbf{0}$

where each of the interval constraints $a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \in [a,b]$ is transformed into the inequalities $a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n + y_{2i-1} \geq a$ and $a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n - y_{2i} \leq b$. If the obtained minimum of $y_1 + \ldots + y_k$ has the value zero,[2] then a solution has been found that does not contain any y_j.[3] Removing the y_j's, the resulting solution vector $\mathbf{x}$ is indeed a feasible solution, i.e., the base is determined to be consistent. If the minimum of $y_1 + \ldots + y_k$ is positive, then the optimal values of the y_j's are larger than zero, i.e. at least one of the y_j's is necessary to keep the base consistent. Since the y_j's were added to the base, the problem itself has no solution. Hence, the base is inconsistent. This forms the algorithm for determining consistency in a decision frame by applying it to the probability and value bases.

Orthogonal Hull

Another important property of a base is the orthogonal hull.[4] According to the definition, in order to calculate the hull, it is necessary to find the pairs $\langle {}^{X}\min(x_i), {}^{X}\max(x_i)\rangle_n$, i.e. finding minima and maxima for single variables in the base. First, a consistent point is determined by employing the procedure above.[5] A search then begins from that point for the minimum and maximum of each variable in turn by forming LP problems with that variable as the objective function. For convexity

[2] It cannot be negative since all y_i's are non-negative by the problem formulation.

[3] Since they are all zero, they can be removed from the problem formulation without altering the solution.

[4] The symmetric hull can easily be subsequently calculated.

[5] That any consistent point is feasible to start with follows from convexity properties of a system of linear inequalities.

reasons, the entire interval between those extremal points is feasible.[6] If the base is consistent, the orthogonal hull can be calculated.

From the two properties consistency and orthogonal hull, most of the other ones in Chapter 4 follow from less demanding computations.

Evaluation Algorithms

The problem addressed in this section is how to compare the different consequence sets computationally using the methods of the previous chapter. The computational pattern that reoccurs several times in that chapter and needs to be solved fast in long sequences is ${}^{PV}\max(\Delta_{ij})$ and ${}^{PV}\min(\Delta_{ij})$. The optimisation of general Δ_{ij}-type of expressions as they appear in Chapter 5 is a demanding computational task as soon as the problem to solve is above toy size. In most cases, however, the expected value rule is employed, making the task less demanding from a computational point of view. In this section, it is assumed that the expected value is being used. Then, the general ${}^{PV}\max(\Delta_{ij})$ turns into ${}^{PV}\max(\sum_k p_{ik} - \sum_k p_{jk})$ for first order Δ-dominance such as 1SE and security levels, and into ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik} - \sum_k p_{jk}\cdot v_{jk})$ for second order ones such as 2SE or NE.

First Order Dominance

For first order dominance, the evaluation expressions are of the form

$$^{P}\max\left(\sum_{k\in K_i} p_{ik}\right) \text{ or } {}^{P}\max\left(\sum_{k\in K_i} p_{ik} - \sum_{k\in K_j} p_{jk}\right) \text{ (or corresponding } {}^{P}\min)$$

for some index sets K_i or index set pairs $(K_i,K_j)(d)$ respectively. These maximisation problems map directly onto LP since it is possible to identify the linear $f(\mathbf{x})$ with $\sum_k p_{ik}$ or $\sum_k p_{ik} - \sum_k p_{jk}$ and note that $\mathbf{Ax} \geq \mathbf{b}$ is the probability base P. The solution to the problem is thus

[6] All convex combinations.

obtained by running a suitable LP solver such as Simplex described later in the chapter. This is an efficient solution to first order problems.

Second Order Dominance

For second-order dominance, the expressions are more complicated. They involve non-linear elements in the form of bilinear terms $p_{ik} \cdot v_{ik}$. The optimisation problems $^{PV}\max(\sum_k p_{ik} \cdot v_{ik})$ and $^{PV}\max(\sum_k p_{ik} \cdot v_{ik} - \sum_k p_{jk} \cdot v_{jk})$ cannot be solved by a simple application of an LP solver even if the P- and V-bases are independent and still consist of only linear expressions. The objective function is $\sum_k p_{ik} \cdot v_{ik} - \sum_k p_{jk} \cdot v_{jk} = p_{i1} \cdot v_{i1} + p_{i2} \cdot v_{i2} + \ldots + p_{im_i} \cdot v_{im_i} - (p_{j1} \cdot v_{j1} + p_{j2} \cdot v_{j2} + \ldots + p_{jm_j} \cdot v_{jm_j})$. This is a bilinear expression with all terms of the form $p_{ik} \cdot v_{ik}$. There is one such expression together with many linear inequalities. Thus, it is an optimisation problem with a bilinear objective function and a system of linear inequalities as constraints. It will be called a bilinear programming problem with ±1 term constants (a BLP1 problem for short).

Four alternative algorithms for use in an interactive environment are proposed here. The bilinear objective function is an instance of quadratic objective functions, and thus the general BLP1 is solvable with quadratic programming methods. The first one, QB-Opt, is the most general, able to solve all BLP1 problems, but being based on QP (see below) it is not as fast as desired for interactive use. The other three are LP-based or simpler and are well-suited for user interaction. The four algorithms are collectively referred to as the B-Opt algorithms. The algorithms are presented in reverse runtime order, i.e. starting with the most general and then continuing with the more specialised ones. Since the bilinear objective function is quadratic, the first natural candidate area for a solver algorithm is quadratic programming.

Quadratic Programming

The theory of quadratic programming (QP) can be found in any standard textbook on non-linear optimisation. Here, only the top-level procedure for searching quadratic optima is considered. The general QP problem with both equalities and inequalities in the constraints is

(QPI) $\max (\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x})$
when $\mathbf{A}\mathbf{x} \geq \mathbf{b}$

where $\mathbf{A}$ is a $m \times n$ matrix with linearly independent rows, $\mathbf{Q}$ is a symmetric $n \times n$ matrix, and $\mathbf{c}$ is a vector in R^n. The expression $\mathbf{x}^T\mathbf{Q}\mathbf{x}$ is a quadratic form, and can contain all possible quadratic terms.

Since the objective function is quadratic, the theory of linear programming as discussed above does not apply. Even though a method similar to Simplex was originally devised by Danzig and Wolfe to solve QP, most methods today use factorised matrices. For any given solution the inequality problem QPI can be considered a problem with only equalities (QPE), namely all weak inequalities satisfied without slack.[7] Since the other inequalities are not active at that solution point they need not be considered locally. This reasoning leads to the active set strategy, a well-known technique within non-linear programming. One of the problems with the active set is that its members at any given step are hard to determine in advance. This means resorting to a guessing strategy, where a choice is made without enough information and corrected later on should the choice be proven unsuitable. QPE problems can be solved using a number of standard methods such as Lagrange methods or null-space methods, depending on matrix sparsity, stability requirements, and other criteria [L89]. The BLP1 problem maps well onto QPI since there is one second-order non-linear expression as the objective function and a larger number of linear constraints in the probability and value bases. The bilinear objective function is a special

[7] An inequality is satisfied without slack if inequalities such as ≥ can be replaced with = and the statement still remains valid.

case of a quadratic function where most of the entries in the **Q** matrix are zero. This forms the basis for the general QB-Opt algorithm.

> **Observation:** Given a decision frame $\langle C,P_3,V_3\rangle$, ${}^{PV}\max(\delta_{ij})$ = max ($\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x}$) with δ_{ij} as $\mathbf{x}^T\mathbf{Q}\mathbf{x}$, **0** as $\mathbf{c}^T\mathbf{x}$ and PV as $\mathbf{A}\mathbf{x} \geq \mathbf{b}$.

The QPE is computationally fairly demanding, and QPI, being an iterative sequence of QPEs, is even more so. Since QPI often does not admit interactive response times, it would be preferable to use an LP-based solver instead. This is possible in a number of important cases, and any of the below algorithms (PB-Opt, VB-Opt, NB-Opt) is preferred, should their preconditions apply, since they invoke LP zero or one time for the solution of a BLP1. Together with QB-Opt, they form a solver hierarchy.

Probability Bilinear Optimisation

The first LP-based algorithm described is the probability bilinear optimisation, PB-Opt. For ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik})$ it solves the general BLP1 problem for $\langle C,P_3,V_2\rangle$-frames while for ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik} - \sum_k p_{jk}\cdot v_{jk})$ it solves all cases where there are no comparative constraints between the consequence sets involved in the calculation, either directly or indirectly. To begin with, maximal and minimal expressions of probability are introduced.

> **Definition 6.1:** Given a decision frame $\langle C,P,V\rangle$,
>
> ${}^{V}E_i^{max}$ is $\sum_{k=1}^{m_i} p_{ik}\cdot b_{ik}$, where $b_{ik} = {}^{V}\max(v_{ik})$.
>
> ${}^{V}E_j^{min}$ is $\sum_{k=1}^{m_j} p_{jk}\cdot b_{jk}$, where $b_{jk} = {}^{V}\min(v_{jk})$.
>
> ${}^{V}\delta_{ij}$ is ${}^{V}E_i^{max} - {}^{V}E_j^{min}$.

The last difference was formed from two linear expressions in only probability variables. The main proposition for PB-Opt is now stated as follows.

Proposition 6.1: Given a decision frame $\langle C,P_3,V_2\rangle$. If none of the comparative constraints in V involve variables from different C_i's, then ${}^{PV}\max(\delta_{ij}) = {}^{P}\max({}^{V}\delta_{ij})$ for any pair C_i and C_j.[8]

Proof: Let $(b_{i1},\ldots,b_{im_i})$ and $(b_{j1},\ldots,b_{jm_j})$ be as in the definitions of ${}^{V}E_i^{max}$ and ${}^{V}E_j^{min}$ above. For all feasible vectors[9] $(p_{i1},\ldots,p_{im_i})$, $(p_{j1},\ldots,p_{jm_j})$, $(v_{i1},\ldots,v_{im_i})$, and $(v_{j1},\ldots,v_{jm_j})$ ${}^{V}E_i^{max} \geq \sum_k p_{ik}\cdot v_{ik}$ and ${}^{V}E_j^{min} \leq \sum_k p_{jk}\cdot v_{jk}$.[10] It follows from $b_{ik} = {}^{V}\max(v_{ik})$ and $b_{jk} = {}^{V}\min(v_{jk})$ and from $p_{ik} \geq 0\ \forall\ k \in \{1,\ldots,m_i\}$ and $p_{jk} \geq 0\ \forall\ k \in \{1,\ldots,m_j\}$. This implies ${}^{V}\delta_{ij} \geq \sum_k p_{ik}\cdot v_{ik} - \sum_k p_{jk}\cdot v_{jk}$. C_i contains m_i consequences. Given two integers $1 \leq k,l \leq m_i$, assume $b_{ik} = {}^{V}\max(v_{ik})$. Then for v_{il}, either (i) there is no comparison $v_{il} - v_{ik} \in [a,b]$ in V, in which case v_{il} is independent of v_{ik}, or (ii) there is a comparison $v_{il} - v_{ik} \in [a,b]$. For case (ii), the constraint can be written (ii a) $v_{il} \geq a + v_{ik}$ and (ii b) $v_{il} \leq b + v_{ik}$. In (ii a) v_{ik} does not constrain the maximisation of v_{il}, and in (ii b) $v_{ik} = b_{ik}$ maximises v_{il}. Thus v_{ik} and v_{il} can be independently maximised and $(b_{i1},\ldots,b_{im_i})$ is a feasible vector as is $(b_{j1},\ldots,b_{jm_j})$ by a similar argument. Since there are no constraints $v_{ik} - v_{jl} \in [c,d]$ in V for different C_i and C_j, each b_{ik} in $(b_{i1},\ldots,b_{im_i})$ and each b_{jk} in $(b_{j1},\ldots,b_{jm_j})$ can be chosen within a consequence set independently of the other sets. ■

This justifies the basis for the PB-Opt algorithm. The rest of the algorithm almost suggests itself. It searches for the optimum ${}^{P}\max({}^{V}\delta_{ij})$ by means of an LP algorithm such as Simplex. The proposition then guarantees that ${}^{PV}\max(\delta_{ij})$ can be determined by calculating ${}^{P}\max({}^{V}\delta_{ij})$ instead provided the precondition is met. Similarly, ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik})$

---

[8] If a graph is constructed with the value variables as nodes and the comparative constraints as edges, then it suffices that there is no path between any variables in C_i and C_j.

[9] Feasible vectors refer to projections from actual solution vectors of the constraint set PV to subspaces.

[10] In order to convey the idea of the proof rather than to obscure it with details, no distinction is made between linear expressions (such as ${}^{V}E_i^{max}$) and instantiations (such as $\sum_k p_{ik}\cdot v_{ik}$).

can be found by searching for an LP solution instead, and in this case there is not even a precondition. Thus, it is a versatile algorithm in the DELTA context.

Example 6.1: Suppose there is a probability base P and a value base V with the following constraints for the consequence sets C_1 and C_2 having three and two consequences respectively.

$p_{11} \in [0.10, 0.40]$
$p_{12} \in [0.25, 0.45]$
$p_{21} \in [0.20, 0.50]$
$v_{11} \geq v_{12}$
$v_{11} \in [0.40, 0.70]$
$v_{13} \in [0.75, 0.85]$
$v_{21} \in [0.30, 0.55]$
$v_{22} \in [0.65, 0.90]$

Now, ${}^{V}E_1^{max} = p_{11}\cdot 0.70 + p_{12}\cdot 0.70 + p_{13}\cdot 0.85$ and ${}^{V}E_2^{min} = p_{21}\cdot 0.30 + p_{22}\cdot 0.65$.

Next, ${}^{V}\delta_{12} = {}^{V}E_1^{max} - {}^{V}E_2^{min} =$
$p_{11}\cdot 0.70 + p_{12}\cdot 0.70 + p_{13}\cdot 0.85 - p_{21}\cdot 0.30 - p_{22}\cdot 0.65$ and
${}^{V}\delta_{21} = {}^{V}E_2^{max} - {}^{V}E_1^{min} =$
$p_{21}\cdot 0.55 + p_{22}\cdot 0.90 - p_{11}\cdot 0.40 - p_{12}\cdot 0.00 - p_{13}\cdot 0.75$.

Finally, ${}^{P}mid({}^{V}\delta_{12}) = ({}^{P}max({}^{V}\delta_{12}) - {}^{P}max({}^{V}\delta_{21}))/2 =$
$({}^{P}max(p_{11}\cdot 0.70 + p_{12}\cdot 0.70 + p_{13}\cdot 0.85 - p_{21}\cdot 0.30 - p_{22}\cdot 0.65) -$
${}^{P}max(p_{21}\cdot 0.55 + p_{22}\cdot 0.90 - p_{11}\cdot 0.40 - p_{12}\cdot 0.00 - p_{13}\cdot 0.75))/2 =$
$((0.10\cdot 0.70 + 0.25\cdot 0.70 + 0.65\cdot 0.85 - 0.50\cdot 0.30 - 0.50\cdot 0.65) -$
$(0.20\cdot 0.55 + 0.80\cdot 0.90 - 0.40\cdot 0.40 - 0.45\cdot 0.00 - 0.15\cdot 0.75))/2 =$
-0.1175. ■

Value Bilinear Optimisation

To circumvent the problem with comparative value constraints between consequence sets while still running an LP-based solver, the value bilinear optimisation VB-Opt is suggested. It solves the BLP1 problem for $\langle C,P_1,V_3\rangle$-frames with the alt-order precondition (defined below) for ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik})$ as well as for ${}^{PV}\max(\sum_k p_{ik}\cdot v_{ik} - \sum_k p_{jk}\cdot v_{jk})$.

Definition 6.2: Given a decision frame $\langle C,P,V\rangle$,

${}^{P}E_i^{max}$ is $\sum_{k=1}^{m_i} a_{ik} \cdot v_{ik}$, where $a_{ik} = {}^{P^k}\max(p_{ik})$ and

P^k is $P \cup \{p_{i(k-1)} = a_{i(k-1)}\} \cup \ldots \cup \{p_{i1} = a_{i1}\}$.

${}^{P}E_j^{min}$ is $\sum_{k=1}^{m_j} a_{jk} \cdot v_{jk}$, where $a_{jk} = {}^{P^k}\min(p_{jk})$ and

P^k is $P \cup \{p_{j(k-1)} = a_{j(k-1)}\} \cup \ldots \cup \{p_{j1} = a_{j1}\}$.

${}^{P}\delta_{ij}$ is ${}^{P}E_i^{max} - {}^{P}E_j^{min}$.

Assume that V is consistent. Then C_i is *alt-ordered* **iff** for all $v_{i1},\ldots,v_{im_i}$, $v_{ik} < v_{il}$ is inconsistent with V when $k < l$.
V is alt-ordered if all C_i, $i \in \{1,\ldots,m\}$, are alt-ordered.

Being able to compare value constraints between consequence sets while still relying on a straightforward LP solution is a fairly strong algorithm property. This must be paid for by the introduction of a restriction on the constraints allowed. For VB-Opt, the restriction mandates that the value base V is alt-ordered.

Proposition 6.2: Given a decision frame $\langle C,P_1,V_3\rangle$ with alt-ordered consequence sets C_i and C_j. Then ${}^{PV}\max(\delta_{ij}) = {}^{V}\max({}^{P}\delta_{ij})$.

Proof: Let $(a_{i1},\ldots,a_{im_i})$ and $(a_{j1},\ldots,a_{jm_j})$ be as in the definitions of ${}^{P}E_i^{max}$ and ${}^{P}E_j^{min}$ above.[11] For all feasible vectors $(p_{i1},\ldots,p_{im_i})$, $(p_{j1},\ldots,p_{jm_j})$, $(v_{i1},\ldots,v_{im_i})$, and $(v_{j1},\ldots,v_{jm_j})$ ${}^{P}E_i^{max} \geq \sum_k p_{ik}\cdot v_{ik}$ and ${}^{P}E_j^{min} \leq \sum_k p_{jk}\cdot v_{jk}$. It follows from $v_{ik} \geq 0\ \forall\ k \in \{1,\ldots,m_i\}$ and $v_{jk} \geq 0\ \forall\ k \in \{1,\ldots,m_j\}$ and from the following argument.

[11] The footnotes in Proof 6.1 apply here as well.

First, $a_{i1} = {}^{P}\max(p_{i1})$ is constrained only by P. Next, for some $k \in \{2,\ldots,m_i\}$, assume a_{ih} (h=1,…,k–1) have been obtained by $a_{ih} = {}^{P^h}\max(p_{ih})$ (P^n is $P \cup \{p_{i(n-1)} = a_{i(n-1)}\} \cup \ldots \cup \{p_{i1} = a_{i1}\}$) and that for some a_{ik} the statement $p_{ik} = a_{ik}$ is consistent with P^k. If $p_{ik} = a_{ik} + \varepsilon$ ($\varepsilon > 0$) is consistent with P^k, then because of the normalisation there are some $p_{ih_1},\ldots,p_{ih_n}$ (where $h_1,\ldots,h_n > k$) such that $\{p_{ih_1} = a_{ih_1} - \varepsilon_1\} \cup \ldots \cup \{p_{ih_n} = a_{ih_n} - \varepsilon_n\}$ is consistent with P^k (where $\varepsilon = \varepsilon_1 + \ldots + \varepsilon_n$, $\varepsilon_i \geq 0$). Since p_{ik} is restricted to occurring in only one compound constraint (the normalisation), the alt-ordering implies that $\varepsilon \cdot v_{ik} \geq \varepsilon_1 \cdot v_{ih_1} + \ldots + \varepsilon_n \cdot v_{ih_n}$. Thus, increasing a_{ik} by an amount ε gives at least as large a contribution to $\sum_k p_{ik} \cdot v_{ik}$ as increasing $a_{ih_1},\ldots,a_{ih_n}$ by a total amount of ε. Thus, ${}^{P}E_i^{max}$ is an optimal way of choosing p_{ik}'s for maximisation. A similar argument applies to ${}^{P}E_j^{min}$.

This implies ${}^{P}\delta_{ij} \geq \sum_k p_{ik} \cdot v_{ik} - \sum_k p_{jk} \cdot v_{jk}$.

In a P_1-base, there are no dependencies between consequence sets. Thus each a_{ik} in $(a_{i1},\ldots,a_{im_i})$ and each a_{jk} in $(a_{j1},\ldots,a_{jm_j})$ can be chosen within a consequence set independently of the other sets. ■

In the preconditions, a probability base of type P_1 is required. This is somewhat over-restrictive since certain P_3-bases can be allowed as well. It is sufficient to require that each probability variable occurs in at most one compound constraint *in addition to* the normalisation constraint and the range constraints.[12] Then the alt-ordering implies that $\varepsilon \cdot v_{ik} \geq \varepsilon_1 \cdot v_{ih_1} + \ldots + \varepsilon_n \cdot v_{ih_n}$ still holds as in the original proof. Increasing a_{ik} by ε gives at least as large a contribution as increasing $a_{ih_1}, \ldots, a_{ih_n}$ by a total amount of ε. This is the key, and the rest of the proof is unaltered.

Example 6.1 (cont'd): Reconsider the previous example. ${}^{V}\mathrm{mid}({}^{P}\delta_{12})$ is calculated in the same manner as ${}^{P}\mathrm{mid}({}^{V}\delta_{12})$ above. The value base is alt-ordered, since $v_{11} \geq v_{12}$ by an explicit expression, and $v_{13} \geq v_{11}$ by non-overlapping ranges. Similarly, $v_{22} \geq v_{21}$ by non-overlap. If, on

[12] The compound constraint may only contain variables from the same consequence set.

the other hand, $v_{13} \in [0.65, 0.85]$ the consequence set C_1 would not be alt-ordered, since the ranges of v_{11} and v_{13} would then indeed overlap. ■

Example 6.2: Suppose there is a probability base P with the following constraints for the consequence set C_1.

$p_{11} + p_{12} = 0.30$
$p_{13} \in [0.00, 1.00]$
$p_{14} \in [0.00, 1.00]$

Suppose there is also a value base V with the following constraints for the consequence set C_1.

$v_{11} = 0.90$
$v_{12} = 0.80$
$v_{13} = 0.70$
$v_{14} = 0.10$

Let δ_{10} denote $\sum_k p_{1k} \cdot v_{1k}$. Since for the simplicity of hand calculations most constraints are equalities, ${}^{PV}\max(\delta_{10})$ and ${}^{V}\max({}^{P}\delta_{10})$ can easily be determined.

Using the definition of ${}^{P}\delta_{ij}$ the expression $a_{11} \cdot v_{11} + \ldots + a_{14} \cdot v_{14}$ becomes $0.30 \cdot v_{11} + 0.00 \cdot v_{12} + 0.70 \cdot v_{13} + 0.00 \cdot v_{14}$.

${}^{PV}\max(\delta_{10}) = {}^{PV}\max(p_{11} \cdot v_{11} + \ldots + p_{14} \cdot v_{14}) =$
$0.30 \cdot 0.90 + 0.70 \cdot 0.70 = 0.76$

${}^{V}\max({}^{P}\delta_{10}) = {}^{V}\max(a_{11} \cdot v_{11} + \ldots + a_{14} \cdot v_{14}) =$
$0.30 \cdot 0.90 + 0.70 \cdot 0.70 = 0.76$

As expected from the proposition, the two values are the same. ■

The following example demonstrates that the proposition above does not imply ${}^{V}\max({}^{P}\delta_{ij}) = {}^{PV}\max(\delta_{ij})$ in general.

Example 6.3: Suppose there is almost the same probability base P with the following larger set of constraints for consequence set C_1.

$p_{11} + p_{12} = 0.30$
$p_{11} + p_{13} = 0.40$
$p_{13} \in [0.00, 1.00]$
$p_{14} \in [0.00, 1.00]$

Suppose there is a value base V with the same constraints for consequence set C_1 as in the previous example.

$v_{11} = 0.90$
$v_{12} = 0.80$
$v_{13} = 0.70$
$v_{14} = 0.10$

To compare with the previous example, calculate $^{PV}\max(\delta_{10})$ and $^{PV}\max(\delta_{10})$ for this base. Using the definition of $^{P}\delta_{ij}$ the expression $a_{11}\cdot v_{11} + \ldots + a_{14}\cdot v_{14}$ becomes $0.30\cdot v_{11} + 0.00\cdot v_{12} + 0.10\cdot v_{13} + 0.60\cdot v_{14}$.

$^{PV}\max(\delta_{10}) = {}^{PV}\max(p_{11}\cdot v_{11} + \ldots + p_{14}\cdot v_{14})$
$= 0.30\cdot 0.80 + 0.40\cdot 0.70 + 0.30\cdot 0.10 = 0.61$

$^{V}\max(^{P}\delta_{10}) = {}^{V}\max(a_{11}\cdot v_{11} + \ldots + a_{14}\cdot v_{14})$
$= 0.30\cdot 0.90 + 0.10\cdot 0.70 + 0.60\cdot 0.10 = 0.40$

Not surprisingly, since the precondition is violated, the two values are not the same this time. ■

The last example shows that if a probability variable is included in more than one compound constraint except for the normalisation, the above proposition might not produce the correct value. The bilinear optimisation algorithm VB-Opt nearly suggests itself. It searches for the optimum $^{V}\max(^{P}\delta_{ij})$ by means of the Simplex algorithm described below. The theorem then guarantees that $^{PV}\max(\delta_{ij})$ can be determined by calculating $^{V}\max(^{P}\delta_{ij})$ instead, using only linear programming techniques.

Restricted Bilinear Optimisation

Looking at the preconditions for PB-Opt and VB-Opt, they do their task of eliminating the need for computational optimisation in one of the bases by exploiting the structure inherent in the input material. The two sets of preconditions do not intersect and they work on separate parts of the frame, one base each. This implies that both preconditions can be combined, forming their union, and then eliminate the need for computational maximisation in either base, in effect abolishing LP from the calculations. This is the key idea of the fourth and last algorithm in the stack. The name, NB-Opt, comes from not needing to run any LP solver at all.

> **Definition 6.3:** Given a decision frame ⟨C,P,V⟩,
> let ${}^{V}E_i^{max}$ be as in Definition 6.1.
>
> Then ${}^{PV}E_i^{max}$ is $\sum_{k=1}^{m_i} a_{ik} \cdot b_{ik}$, where $a_{ik} = {}^{P^k}\max(p_{ik})$ and
> P^k is $P \cup \{p_{i(k-1)} = a_{i(k-1)}\} \cup \ldots \cup \{p_{i1} = a_{i1}\}$.
>
> Also let ${}^{V}E_i^{min}$ be as in Definition 6.1.
>
> ${}^{PV}E_i^{min}$ is $\sum_{k=1}^{m_j} a_{jk} \cdot b_{jk}$, where $a_{jk} = {}^{P^k}\min(p_{jk})$ and
> P^k is $P \cup \{p_{j(k-1)} = a_{j(k-1)}\} \cup \ldots \cup \{p_{j1} = a_{j1}\}$.
>
> Then ${}^{PV}\delta_{ij}$ is ${}^{PV}E_i^{max} - {}^{PV}E_j^{min}$.

The proofs of PB-Opt and VB-Opt apply to one part each of the proposition below. They are independent as are the preconditions, thus they can be joined together. Their union justifies NB-Opt.

> **Proposition 6.3:** Given a decision frame $\langle C,P_1,V_2\rangle$, assume that none of the comparative constraints in V involve variables from different C_i's. Further, suppose that the consequence sets C_i and C_j are alt-ordered. Then ${}^{PV}\max(\delta_{ij}) = {}^{PV}\delta_{ij}$ for any pair C_i and C_j.

The algorithm based on this proposition is very fast since no pivoting procedure needs to be invoked if the hull has been pre-determined. NB-Opt is the ideal first algorithm to run in an anytime algorithm stack since an approximate answer can be supplied almost instantaneously.

The Solver Stack

The four algorithms together make up the B-Opt solver stack. This stack has the property that one of the faster algorithms can be selected to run first in order to receive an approximation. The approximation error is then corrected as the appropriate solver is subsequently allowed to run. Figure 6.1 shows the solver stack referral chain for approximate or anytime computations.

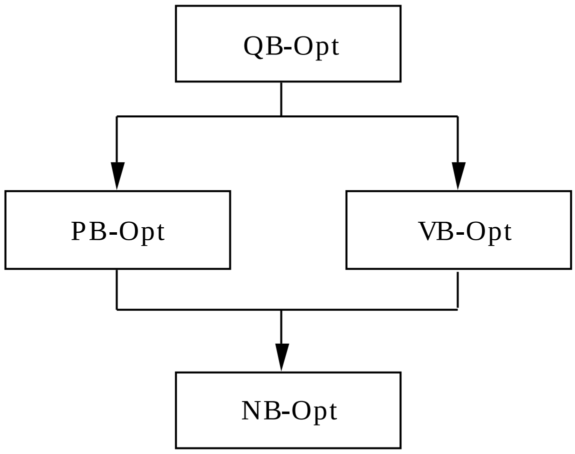

Figure 6.1 The B-Opt referral chain

Thus, the stack forms an anytime algorithm, with the property of delivering a reasonable answer if being prematurely aborted [Z96]. This is a convenient property in interactive applications.

The Simplex Method

The algorithm for finding the orthogonal hull relies on the ability to solve a sequence of small LP (SSLP) problems rapidly. The frame evaluation requires higher-level algorithms that generate long SSLP sequences to an even greater extent. The most appropriate candidate for an SSLP solver implementation is the Simplex method.

From the early 1950s onwards, the very general nature of the LP problem formulation rapidly led to the solution of an increasing number of ever larger problems in industry and government. With the growth of computing in general, the area of LP soon gained momentum. The Simplex algorithm, originally suggested by Danzig in 1947, is one of the earliest solution methods. At first, it was not much more than a clever way to manipulate matrices in order to manœuvre from one corner to another of a feasible polytope in such a way that the objective function never decreases. Today it has become an entire sub-field within applied mathematics. The current research focus is on solving larger and larger problems, involving thousands of inequalities and tens of thousands of variables. The employed techniques are in some respects akin to research in numerical methods [G92].

Problems still remain with the Simplex method. A theoretical problem is that it belongs to the class of exponential algorithms in time. Examples can be designed to reveal this deficiency, see for example [C83].[13] Because of this, other, non-linear approaches to LP problems have been suggested, notably Khachian's ellipsoid method[14] and the Karmarkar algorithm. The proposed advantages of these non-linear approaches only reveal themselves in very large or contrived problems. It is evident from recent research summaries that almost all research focuses on solving large LP (LLP) problems within a reasonable time[15] [G92, W91]. As was pointed out above, the task in DELTA is to find solutions to a sequence of small problems in a short time to allow for interactive use.[16] None of the non-linear methods, nor much of the current research in Simplex is therefore of any great use in this thesis. Some extensions to the standard Simplex algorithm are examined to see whether they can contribute to the development of a fast algorithm for SSLP problems. Other Simplex techniques were discarded because they apply to specially structured or very large problems, and many were related to numerical properties of very large matrices. The descriptions

[13] While the general LP problem is polynomial, the algorithms in this chapter are based on the Simplex method. Thus, they will inevitably be classified as exponential in time. In real-life applications, Simplex performs very well, and there is no reason to expect any less from algorithms based on it.

[14] Khachian did not invent the method but provided a proof that it is polynomial.

[15] Which might mean hours or even days.

[16] Typical Simplex execution times are less than a second for a 100×100 problem and less than a minute for a 1000×1000 problem, scaling as $O(n^2)$.

of the extensions given here are intended to be intuitive for the purpose of arguing for and against their inclusion in DELTA solver algorithms.

Revised Simplex

In each Simplex step, one basic solution is replaced by another by means of matrix operations on the coefficient matrix **A** and the right-hand side **b**. If the size of **A** is $m \times n$, then a Simplex solution to an LP problem can most often be found in $3 \cdot m/2$ steps, each step including a pivot operation consisting of a large number of multiplications and divisions [L89]. Most LLP problems have a structure where $m << n$ and only a minor fraction of the columns will ever be pivoted on. Because of this, it seems to be a waste of processing time to update all columns in every step. Using matrix algebra, it can easily be shown that the column to pivot on in each step can be constructed from the original data instead of from the data in the previous step. All potential transformations are held in a matrix, and the total amount of processing of columns is now proportional to m instead of n, but an overhead penalty is incurred for keeping track of the dormant columns. If $m << n$, as in the LLP problems of mainstream Simplex research, then this is a very large improvement. However, in the SSLP case, $m \approx n$. Both methods iterate the same number of steps, but since a large fraction of the columns will be used actively, the overhead introduced in the revised method makes it less attractive than the standard method for SSLP purposes. The revised formulation of the Simplex method is not applied to the DELTA solvers.

Upper and Lower Bounds

In many LP problems, a considerable number of the constraints have only one variable, reflecting a modelling situation where there are many constraints on single variables, in some cases on most of the variables involved. It means that if there were a formulation of Simplex where these constraints could be handled in an efficient way, the computational effort for solving the problem could be greatly reduced. This is due to the fact mentioned above that the effort expended on solving an LP problem is roughly proportional to $3 \cdot m/2$, where m is the number of constraints.[17] Since constraints on single variables are still matrix rows[18] they account for a fair amount of the computational processing of such problems.

Suppose that the variable x_i is subject to the constraints $x_i \geq a_i$ and $x_i \leq b_i$.[19] In the standard formulation, this would be introduced into the problem in the form of two constraint inequalities, i.e. two matrix rows, increasing the m above by two. Instead, by the formulation of the LP problem, all x_i's are automatically subject to the constraint $x_i \geq 0$. The variable x_i is then transformed into $x_i' = x_i – a_i$ and the coefficient matrix and objective function are adjusted accordingly. The new variable x_i' is now subject to the constraints $x_i' \geq 0$ and $x_i' \leq b_i – a_i$, which eliminate the need for an explicit lower bound.

For the upper bound, the reasoning is only slightly more involved. By defining $x_i'' = x_i' – (b_i – a_i)$, and substituting one for the other back and forth during the Simplex execution, a variable at its upper bound can be regarded as non-basic. When the variable x_i' reaches its upper bound during a Simplex iteration step, it is replaced by x_i'' and vice versa. Then the new variable is zero by definition and becomes non-basic.[20] Thus the implicit constraint $x_i' \geq 0$ (or $x_i'' \geq 0$) is again used to eliminate the need for an explicit row entry in the coefficient matrix. This is highly applicable for the DELTA solvers since there are upper and lower bounds on almost every variable in a base.

17 Inequalities or rows in the coefficient matrix.

18 With only one non-zero coefficient.

19 $x_i \in [a_i, b_i]$ in interval notation.

20 Do not confuse the Simplex concept of base, meaning non-zero variables, with the DELTA concept of base, meaning a collection of interval constraints.

Generalised Upper Bounds

There is a promising generalisation of the upper bound handling in the previous paragraph. Some LLP problems have a structure where many constraints are of the form $\sum_i x_i = b$ for non-trivial index sets. There is a close relationship with the probability base where the normalisation equation is $\sum_k p_{ik} = 1$ for each consequence set. The theory of generalised upper bounds (GUBs) is a matrix method based on factorising the base into parts with different properties. The new parts are then less complicated to solve. Suppose the coefficient matrix has m rows of which m_2 are of the generalised form above. The GUB technique is then reported to become faster than ordinary revised Simplex when $m_2 \approx 0.3 \cdot m$ and ten times faster when $m_2 \approx 0.8 \cdot m$ [C83]. While this is a remarkable speed increase for real LLP GUB problems, there is only one normalisation equation per consequence set in the probability base,[21] and that falls well below the trade-off point, making this extension unimportant.

Implicit Identity Matrix

The implicit identity matrix technique is a simple observation of how the Simplex algorithm works. In any matrix description of the standard Simplex, it is readily seen that the basic variables (i.e. those with non-zero values assigned) form an identity sub-matrix within the coefficient matrix. Since this is an invariant fact during the entire Simplex execution, that part of the matrix might as well be replaced with index values in a vector. The problems considered here are not very large, and so the trade-off should be balanced between program code for treating special cases and savings in memory space and numerical operations. It is also easy to combine with the sparse matrix encoding below. The outcome depends on the architecture of the executing machine but the gain or loss is not very substantial for the SSLPs. A longer discussion of architectural impacts on implemented algorithms can be found below.

[21] And none in the value base.

Sparse Matrix Encoding

For LLP problems, the matrices often become very large. An ordinary LLP problem might have 1,000 rows and 10,000 variables and this would result in 10^7 matrix elements, most of which contain zero values. Obviously, this is unfeasible to handle. By observing that only a small fraction of the elements in each row are non-zero, the Simplex algorithm can be modified to work with a one-dimensional structure representing only the non-zero elements of the coefficient matrix. All elements not found in the structure are zero by definition. Extra program code is required to handle this, but the processing overhead is small compared to the savings in memory and increase in speed achieved for LLP problems. SSLP problems do not gain as much from sparse matrix techniques, since each matrix is rather small. They are not as sparse as LLP matrices[22] but the approach is still of importance. There is one circumstance that is especially important. If the architectural speed of floating point (FP) operations is much slower than testing integer and pointer vectors, then sparse matrices are of extra interest, but in that case this is a special case of the FP speed issue below and is included in the hardware trade-off problem.

Sensitivity Tests

An important part of the Simplex method is the provision of convenient means to do sensitivity analysis without reworking the problem, but rather by reasoning about small differences in the input data. There are standard reasoning patterns for carrying out sensitivity analysis of the attained optimal solution. In this way it is possible to vary the coefficients of the objective function or the right-hand side to see within which ranges the respective coefficients can vary while still keeping the same solution as optimal.[23] Unfortunately, this does not

[22] Remember that single variable constraints are handled by the upper and lower bound technique.

[23] Even though the optimal value may change.

map very well onto either the consistency or the orthogonal hull problems. To see this, suppose that the proposed algorithm arrives at a solution to the problem

$\min (y_1 + \ldots + y_k)$

when $\mathbf{Ax} \geq \mathbf{b}$

and $\mathbf{x} \geq \mathbf{0}$, $\mathbf{y} \geq \mathbf{0}$,

and inquire whether this minimal value is zero or not. Usually, there are many combinations of basic variables that achieve this because there are many possible feasible basic solutions. The Simplex sensitivity analyses focus on properties of the obtained basic configuration, while here any solution (of the often many) with the desired property is accepted. Thus, Simplex sensitivity reasoning is of little value to the solver. Instead, the DELTA sensitivity analyses take place on a higher level, using the concepts of expansion and contraction.

The Dual Problem

An important theoretical as well as practical issue is the theory of duality. For each optimisation problem, linear or not, there exists another problem called the dual problem, which represents the strongest possible relaxation of the primal problem. The details of this theory are omitted here. One of its immediate LP applications is that in order to solve an $m \times n$ LP maximisation problem, it is equally effective to solve a dual $n \times m$ minimisation problem. While a minimisation and a maximisation problem present the same computational load, the dual problem is more interesting to solve if $m > n$, in which case the dual problem contains fewer rows and, as mentioned earlier, rows account for most of the processing time. The SSLP problems do not fit this description well, so the duality techniques are left unexplored.

Implementation

Unfortunately, these Simplex calculation techniques are not enough. An empirical investigation into Simplex performance revealed other problems with using Simplex for SSLP tasks [D95]. The problems are hardware instruction-set related and come from differences in the architecture of different computers. It was necessary to develop techniques for handling these problems; otherwise, DELTA would have become less of an interactive method. The following contains a discussion of some implementation issues.

Copy Speed

There are two main classes of operations to perform in a Simplex execution, apart from controlling the program flow. These are copying matrices and performing floating point (FP) arithmetic. Regardless of the implications of theoretical investigations into preferred executing techniques, the relative speed of copy and FP operations has a large impact on the algorithm to be executed. If FP operations are slow compared to copying memory contents, then saving partial results becomes more important. Also, restoring phase one solutions becomes meaningful, since an extra pivot takes longer time than to resume processing on a copy of a previous state. If FP operations are fast compared to copying, the opposite strategy is to prefer. What is meant by fast or slow FP arithmetic is discussed in [D95].

Guarded Operations

For much the same reasons as in the discussion on copy versus FP speed, it is important to guard FP operations on some architectures.[24] This means that for certain FP operations (e.g. multiplication or division) the guard should check for non-effective operations (e.g. multiplying or dividing by one). For example, on a machine where FP arithmetic is much slower than memory comparison, an FP division that

[24] This often occurs on machines that lack FP operations in hardware.

is executed frequently in an inner loop might be ten times slower than a check for a numerical value in the denominator. Since dividing by one is not uncommon in normalising matrix rows, speed improvements are noticeable if the operation is not carried out (i.e. guarded) when the denominator is one. For other architectures, guarding instructions can at best be meaningless, and at worst slow down the Simplex execution.

Extra Cost Rows

In the formulation of LP for calculating the orthogonal hull, remember that besides finding any solution at all (as for consistency), the maximum and minimum points for each variable or constraint must be determined.[25] This amounts to solving $b_i = \max(f(x_i))$ and $a_i = \min(f(x_i))$ for all x_i's. Each max- and min-problem is an LP problem in its own right, but the total speed can be increased substantially by observing that each one of the problems is better off starting from the point determined by the search for a consistent point than starting from scratch. However, this requires that the cost rows for all max- and min-problems are transformed to a consistent point. To achieve this, all cost rows participate in the Simplex iteration steps. Since solving a min- or max-problem modifies the consistent original solution, there are two alternatives. One is to copy the consistent solution before solving each problem, and the other is to continue with the next problem from the point where the most recent solution found its optimal value.[26] The choice depends again on the relative speeds of copy and FP operations.

Empirical Results

In [D95] a number of development environments were measured with respect to execution speed of some critical instructions, notably copying

[25] In the discussion that follows it is assumed (without loss of generality) that all constraints are range constraints. The reasoning applied is the same for compound constraints.

[26] Which probably is a point farther from the next optimum to search for.

memory and performing FP arithmetic. It is there clearly seen that three broad classes of architectures are common, even though there are no sharp, clear-cut speed ranges. Almost certainly, the ranges will change over time. Also in [D95], a number of Simplex techniques were evaluated in typical environments from the three identified different architecture types. Further, measurements were undertaken to support the conclusions in the Simplex discussion above. The concluding observation after the series of experiments was that a configuration program is needed to measure particular operations on the target architecture and compiler in order to set compilation parameters for configuring the source code for optimal execution in that environment.

The three identified classes of architectures were categorised according to their FP hardware. The first category, mainly consisting of workstations, has FP hardware units integrated into the CPU chip or board. The next category, consisting of personal computers with FP support, has hardware co-processors and can carry out FP arithmetic in hardware. The last category contains low-end personal computers lacking FP hardware, and they are thus forced to make all FP calculations in software.

For the three classes, different subsets of the available options proved to be optimal. This was mostly due to differences in the execution speeds of memory copying and FP arithmetic. As the machines measured in the empirical studies are not the only ones available, now or in the future, the best solution to the architectural problem seems to be a configuration program. Such a program would measure the interesting speeds of instructions and set source code parameters accordingly. The source code would then automatically be recompiled prior to execution on a new platform. In this way, the source code becomes independent of the actual target machine. This independence relies on the source code containing all appropriate techniques as inclusion options for the configuration program to choose from. Some complicated interdependencies might render the configured source code non-optimal, but the tailored solver would almost certainly be closer to the optimum than a solver not being configurable.

This concludes the discussion on optimisation algorithms and ends the presentation of the DELTA method in Part II. The last part is the Supplement containing a summary, notes on further research, appendices, references, and an index.

If you should go skating
On the thin ice of modern life
Dragging behind you the silent reproach
Of a million tear-stained eyes

Don't be surprised
When a crack in the ice
Appears under your feet
You slip out of your depth
And out of your mind
With your fear flowing out behind you
As you claw the thin ice

– R. Waters

Conclusion

The thesis ends with this conclusion, two appendices, references and an index. The conclusion contains a summary and some pointers to areas of further research.

Summary

This thesis is about *Computational Decision Analysis*. Each of the three words in the title is a keyword. “Decision” means that it deals with selection problems, i.e. situations in which there are more than one alternative course of action. “Analysis” means that there are no absolute bits of advice given, no single best alternative pointed out by a mechanised procedure, but rather an aid is provided for understanding the decision problem and how the solutions relate to each other. “Computational” means that there exist efficiently computable algorithms that perform the analysis in a reasonably short time in order to admit interactive analysis.

The thesis presents the DELTA method for decision-making using imprecise information. The objective is to describe a method for evaluating choices under uncertainty.[1] The nature of most information available to decision-makers is imprecise, be it information for human managers in organisations or for process agents in a distributed computer environment. In spite of this, most traditional models for decisions disregard this state of affairs. Some more modern approaches, like

[1] Choices under risk in classical decision analysis.

fuzzy decision analysis and Dempster-Shafer-based methods, address the problem of vagueness. Many of these modern approaches concentrate more on representation and less on evaluation. The emphasis in this thesis is more on evaluation, and even though the representation used is that of standard probability theory, the use of other well-established formalisms is not ruled out.

Introduction

The first part introduces decision analysis in general and the DELTA method in particular. *Chapter 1* begins by surveying a number of traditional decision models and discussing some of their properties. The models are divided into three categories: risk-free, strict uncertainty, and risk models. For the latter, some more modern approaches to imprecision in input data are discussed. Finally, appropriate research methods are discussed.

Chapter 2 presents a suggested decision method for human decision-makers in work cycle form based on the DELTA method. It attempts to convey some feeling for how a decision maker can utilise the method in analysing a decision situation. It also tries to demonstrate that the suggested method is realistic to work with.

Chapter 3 presents the DELTA Decision Tool (DDT), an interactive graphical software implementation of the DELTA method intended for aiding human decision-makers in understanding and analysing real-life decision situations. The chapter opens with a description of the DDT software and its architecture. Most of the chapter is devoted to an industrial example, which is used in presenting some of the features of DDT and the user interaction.

Representation

The core of the thesis is the presentation of the DELTA method in Part II. *Chapter 4* starts with the structure of a decision problem and the required representation of user statements. A model of the situation is created with relevant courses of action and their consequences, should specific events occur. The model is represented by a decision frame. The courses of action are called alternatives in the model, and they are represented by consequence sets in the decision frame. Following the establishment of a frame, the probabilities of the events and the values of the consequences can be filled in. All statements should have an interval form to reflect the imprecise nature of the input data. Next, the chapter also presents general properties of bases, i.e. collections of constraints and core intervals. Further, properties particular to bases of probability statements and then the value base counterparts are discussed. Finally, the section on translations shows suggested representations of numerical and qualitative statements of both probability and value.

Evaluation

Chapter 5 presents evaluation methods in detail. The DELTA method is presented step by step, beginning with the discussion of the expected value rule for selection amongst a number of available courses of action. Then a number of other evaluation rules to either replace or supplement the expected value are presented. They are discussed from a choice rather than preference view. One of the conclusions is that there exists no perfect rule, although the expected value seems to be at least as good as many of its contenders. To improve that rule (or any other similar rule), it is suggested that it should be supplemented with other, qualitative rules rather than engaging in further modifications in chase of the perfect rule. A characteristic of qualitative rules is that they do not rely on multiplying probabilities and values but treat them as separate numeric entities. Once a rule has been agreed upon, it can be applied to all the alternatives, provided there is a computational procedure for evaluating the alternatives under that rule. The DELTA dominance is introduced as a unifying concept for many of the dominance rules in current use. Dominance and threshold methods are discussed and the kinship between them is pointed out.

Dealing with imprecise statements means frequently encountering decision situations where more than one alternative is to prefer in different parts of the consistent solution space to the constraints. Consequently, dominance selection rules are not enough to indicate preferred choices. Many ideas have emerged in response to the problem arising when the information given is imprecise and overlaps in the sense that parts of the information seem to favour one alternative (consequence set) while other parts favour another one. This thesis conforms to statistical decision theory and introduces some new concepts to aid the selections. The concepts of maximal and minimal differences represent the most and least favourable possibilities respectively. A new set of selection rules is introduced – the concepts of strong, marked, and weak dominance. The selection procedures suggested are based on those concepts and on the expansion and contraction principles from Chapter 4.

Optimisation

Chapter 6 deals with computations for DELTA, especially optimisation algorithms since they are the most demanding ones. It starts with linear programming (LP) for determining properties of bases. For solving these LP problems computationally, the Simplex method is used. The chapter continues with bilinear programming, necessary to calculate the results of the evaluation rules. First quadratic programming is discussed and then three algorithms are presented that under mild constraints solve the required bilinear programming problem (BLP1) with Simplex techniques instead. Due to the unusual problem structure (a long sequence of smaller problems rather than the usual single large one), each of the Simplex techniques must be carefully considered in order to select which ones to apply. Furthermore, hardware architectural issues are found to be important for the implementation of the DELTA solver.

The conclusion is that a configuration program is necessary, which will measure the relative speeds of different operations and configure the solver's source code accordingly.

Appendices

There are two appendices, each addressing one application area of computational decision analysis. *Appendix A* deals with problems of co-ordinating multi-agent systems and the applicability of DELTA to that area. In dealing with rational agents and their ability to make decisions, it is again emphasised that there is no universal rule with which rationality could be equated. Instead, the conclusion is that a successful agent must be good at analysing results from a set of reasonable decision rules. Such analyses should ideally exploit several decision rules shown appropriate for the particular domain of interest. Agents using the expected value and security levels are discussed in the appendix, but it should be noted that these are not the only possible rules and the method could use other decision rules as well.

Appendix B applies DELTA to the area of risk analysis by introducing the DEEP (Damage Evaluation and Effective Prevention) method. A risk analysis method is presented that substantially improves the evaluative phases compared with other, earlier approaches. The presentation is focused on the analysis and identification of threats and on the evaluation of the suggested actions since those are the steps where the DEEP method differs the most from other methods. The idea behind DEEP is to offer an analytical framework for risk management in the classic chain identification–valuation–action without trying to replace it.

Further Research

The DELTA method should be seen as a framework. Even though the method is in a sense complete, there are numerous plausible research tasks to extend it in several directions. Two of the most important directions are

(i) augmenting the method with new features, and
(ii) extending the method to handle multiple criteria.

There are other directions as well. One is extending DELTA to handle multi-level trees in other ways than the obvious compound strategies mentioned in Chapter 1. Another is generalising security levels to account for other types of undesirable results.

New DELTA Features

The current DELTA method may be augmented by new features along a number of different lines of development including representation, evaluation, and computation. It is also desirable to conduct larger field studies on the real-life use of tools based on DELTA.

Representation

Today DELTA uses probabilities in the form of numbers 0–100%. Another type of input probability is the odds formulation. The odds of an event E is $p(E)/p(\neg E) = p(E)/(1-p(E))$ for some probability function p. Sometimes this is felt to be a more natural way of expressing probabilities for decision makers. It has gained some popularity within probabilistic reasoning in conjunction with using Bayes' Rule, where advantages can be found in not having to specify certain probabilities [GN87]. The odds formulation could be of use for DELTA as well in allowing the input probabilities to be in odds form, should that be found to be more natural. This is an open question but warrants further investigation.

Another input issue is the user interaction in tools for human decision-makers. In DDT, the input is handled using rulers to enter essentially numerical data, while in Chapter 4, there is a section on the translation of linguistic input data. Those two forms may be combined in various ways, for example by extending DDT to handle qualitative statements as well. This poses questions about how to design such an interface with regard to alternative interpretations of vague statements and sensitivity analysis of non-numerical data.

This thesis only considers standard probability systems for representing decision-maker statements. Other approaches mentioned in Chapter 1 include Dempster-Shafer theory and fuzzy decision analysis. These other approaches also allow the decision maker to model and evaluate a decision situation in vague terms, but using other means to deal with vagueness or imprecision. It is plausible, for example, to view the concepts of expansion and contraction as membership functions on fuzzy sets corresponding to interval constraints. It would probably be worthwhile to consider the DELTA evaluation framework for those methods too. The cross-fertilisation would certainly be beneficial for the DELTA evaluation method and possibly for the others as well.

Evaluation

The general Δ-dominance rule is introduced as a unifying concept. In its generic form, it describes the type of dominances to be considered and thus the type and amount of computation involved in evaluating alternatives in the framework. It is very general and many instantiations are possible, of which a few are given in Chapter 5. It would be interesting to further explore the Δ-dominance concept with more rule instances. The classification into dominance orders opens up questions of higher order rules – are they necessary and what are their properties and instances? Certainly, also first- and second-order classes have interesting members not mentioned in the thesis. While general numeric rules have been considered on paper, the only implementations so far are based on the expected value. It is also interesting to study further several different replacement rules and their use in a real-life tool.

The selection procedures are not very precise, partly due to the nature of the decision problem, and partly because the dynamic interconnection between strong/marked/weak dominance and expansion/contraction needs further study. It depends on the decision situation, on whether the decision maker is a human or a machine, and on whether the goal is to make a final decision or to gain a better understanding of the decision problem.

Optimisation

For consistency and hull calculations, Simplex is the most versatile method, but it is more general than required. An important task is to try to develop more specialised algorithms. These algorithms would probably not be pivoting matrix algorithms, but use some special representations that exploit the structure of the problems better.

The general BLP1 problem is today covered by QP, usually solved with factorised matrix techniques. There is another, perhaps more promising technique to solve QP problems. The area of linear complementarity problems (LCP) originates from the mid-1960s, and in the beginning LP and QP were the main application areas of LCP. Those are still important, and the search for first-order optimality conditions in QP problems by means of LCP algorithms is one possible direction for developing a general BLP1 algorithm. See [CPS92] for a thorough discussion of LCP.

There are, however, two main reasons for preferring a different solution to the BLP1 problem. The first has to do with execution times. In larger problems, any algorithm for solving BLP1 will start running too slow for interactive purposes. Then it is desirable to have an anytime algorithm. This means that it can be prematurely asked for a preliminary solution, and this solution should be a good approximation of the exact result to be obtained in time due.

The second reason is connected to future developments of the DELTA method. As indicated below, multiple criteria are a reasonable extension to DELTA. Another extension is layers of credibility and trust in the multi-agent application of Appendix A. In those cases, the objective function will not be bilinear but multilinear, the terms having the form $w_k \cdot p_{ik} \cdot v_{ik}$ if weights are included, $c_k \cdot p_{ik} \cdot v_{ik}$ if credibilities are included, or even more terms if other situations are modelled. Thus, a good BLP1 algorithm should be extendible to handle multi-linear problems. This is not the case for QP algorithms since they solve only quadratic problems, while multi-linear problems might be of higher order. The B-Opt family should be possible to extend to the multi-linear case.

The LP-based bilinear optimisation algorithms are efficient but require certain kinds of decision-maker expressions to be left out, most notably some comparisons between either probability or value variables. While this is not a great problem, the solver stack would be more complete if the range of algorithms available could cover all possible decision frames without resorting to QP techniques, since the full power of the QP formulation is not needed. Such an algorithm could be called PVB-Opt. The bilinear objective function and the constraints are separable into probability and value parts, and the first order differentials have a certain structure that ought to be possible to exploit in a way similar to how linear programming uses its constant differentials.

There are essentially two ways of approaching the design of PVB-Opt. In the first, PB-Opt and VB-Opt are taken as starting points, developing some more advanced but still LP-based algorithm that executes in one of the probability (P) or value (V) bases at a time. This seems promising at first. The constraints in P, V, and $P \cup V$, being systems of linear inequalities, form compact convex sets, but since the objective z is non-linear, the gradient components $\frac{\partial z}{\partial p_{ik}}$ and $\frac{\partial z}{\partial v_{ik}}$ alter their signs in ways that are hard to control and risk winding up in local optima. The other way to design the PVB-Opt algorithm is to start with a general QP algorithm and remove functionality not needed because of the special structure of BLP1. Early QP algorithms, based on LP-style ideas, have means of controlling that issue and ought to be investigated. Which road will lead to the goal, and with what grade of success, is an open question. Naturally, such an algorithm would run slower than the LP-based PB-Opt and VB-Opt but hopefully considerably faster than ordinary QP algorithms.

Empirical Studies

The DELTA method needs more exposure to real-life decision problems. It would be interesting to apply the method to a number of real-life situations and to compare the outcome with unaided decisions made in parallel. This has been done in a NUTEK project together with Banverket (The Swedish National Rail Administration) [DE97b]. Banverket intended to procure railway equipment for around 5 billion SEK and conducted a large evaluation of all prospective suppliers. The results from this empirical study are encouraging but more studies are needed. Preferably, such empirical studies could partly be made in co-operation with researchers from the area of psychology.

Multiple Criteria

A decision can often be seen from different perspectives, usually called criteria, and the expected (or numerical) values of the alternatives are often different when seen from the different criteria. This is the research area Multi-Criteria Decision Analysis (MCDA), see for example [V92] or [B90]. Traditional criteria include finance, environment, policy, public opinion, competence, and growth opportunities.

Example: A chemical industry is about to invest in a new purifying plant for wastewater. This investment decision can be studied using several criteria. The financial criterion is usually the first, where costs incurred from the actual investments as well as losses in production efficiency are considered. The environmental criterion might concern the risk of being sued for polluting or causing other harm to the environment, which could also lead to liabilities and badwill for the company. A further criterion is that of personnel, where care must be taken to prevent noise and chemical hazards, and thought must also be given to changes in employee responsibilities. It is no easy task to evaluate these different criteria manually and then make a total evaluation. Normally, a company spends considerable time, effort, and money on feasibility studies before making such a large investment. ∎

In most cases, the preferred alternative is not the same for all criteria. The alternative with the highest cash flow is seldom the best from an environmental point of view, or seen from the perspective of the employees. How can all these conflicting requirements be taken into consideration?

If there is one criterion that can clearly be considered the most important, then the simplest way would be to evaluate the alternatives only with respect to that criterion. In that case, all the information from other criteria would be disregarded, almost certainly leading to lower quality of the decision. For example, if an alternative is marginally better according to the financial criterion, but much worse from an environmental point of view, then a decision in favour of that alternative could be considered sub-optimal.

The aggregation of utility (or value) functions under a variety of criteria is investigated in the area of Multi-Attribute Utility Theory (MAUT), see for example [K92, KR76, F70]. A number of techniques used in MAUT have been implemented as computer programs such as SMART [E77] and EXPERT CHOICE, the latter being based on the widely used AHP method [S80]. AHP has been criticised in a variety of respects [WF82, BG83] and models using geometric mean value techniques have been suggested instead [BCG87, K87]. Techniques based on the geometric mean value have, for instance, been implemented in REMBRANDT [L93]. All these approaches have their advantages, but as for the probabilistic decision situations treated in this thesis, the requirement to provide numerically precise information sometimes seems to be unrealistic in real-life decision situations. Some multi-criteria models with representations allowing imprecise statements have been suggested. For instance, the system ARIADNE [SW84] allows the decision maker to use imprecise estimates but does not discriminate between alternatives when these are evaluated into overlapping intervals. [SH95] extends the AHP method in this respect and makes use of structural information when the alternatives are evaluated into overlapping intervals.

One research direction is to extend the DELTA method with the capability of handling multiple criteria. One idea would be to assign the different criteria weight factors between 0 and 1 according to their relative importance and require the weights to sum to one. The weights would then not be given as absolute real numbers, but could be in the form of interval statements such as *criterion C has an importance weight between 0.2 and 0.5* or *criterion C is more important than criterion D*. The value base would then be extended with weights and sets of values for each criterion and then evaluated using additive MAUT-type of rules or other types. That is a natural extension of the DELTA method that warrants further research [DE97a].

A

Multi-Agent Systems

This appendix deals with an application area of computational decision analysis, the area of multi-agent systems. The content of the appendix is joint work with Magnus Boman, DSV, and Love Ekenberg, IIASA. The text is partly derived from [EDB96b] and [EDB97].

Distributed AI (DAI) emerged as a research field in its own right around 1980 [BG88] and a partition is often made into distributed problem-solving systems (DPSs) and multi-agent systems (MASs). Both parts of DAI are important to software systems. The DPS part covers the case when a coordinating agent controls a set of agents in order to accomplish some task in a distributed way. The MAS part covers the case when a set of agents must act on their own without immediate aid from a coordinator. In the former, there is a global task that needs to be solved and usually a global notion of utility that can constrain the actions of the intelligent agents. In MASs, by contrast, there is no such global notion of utility [R93].

Theories of intelligent agents offer means for dealing with the complexity inherent in developing distributed systems, and the advances in DAI over the last five years have affected the design methods of distributed software in several ways. One main issue in DAI is how a group of agents can cooperate in order to solve different tasks and how such a system of agents can be coordinated. Some aspects of decision theory have influenced the area of MASs [RS95], partly as a result of philosophical aspects of agent rationality [D92], and partly because of interest in extending the principle of maximising the expected value in efficient real-life applications [B95].

The idea in this appendix is to demonstrate that a method for evaluating reports from sets of autonomous agents from a decision analytical viewpoint can be built around the DELTA concepts. A decision-making agent (DMA) may make use of imprecise and possibly incomplete reports made by different autonomous agents when co-ordinating its activities and deciding which action strategy to adopt. In a manner similar to the standard DELTA method, these reports are translated into a suitable representation and the strategies are evaluated. The set of non-dominated strategies is usually too large after a first evaluation and the situation needs to be analysed with respect to further discriminating principles. To allow the DMA to make a flexible analysis of its decision situation, a method such as the one described here ought to contain the possibility of analysing the situation in several respects. Since DELTA includes efficient evaluation of non-trivial decision problems, the method and implementations thereof are well suited for use in the reasoning mechanisms of more sophisticated agent-based information systems, and it is quite straightforward to include a multitude of decision rules in this framework.

This particular application considers a decision problem with respect to the contents and the credibilities of the received reports. These two aspects are modelled in an agent decision frame consisting of two systems of translated interval statements, similar to an ordinary decision frame. Once it is decided that a set of agents should achieve some goal, and some semantic mapping has been provided for any syntactically heterogeneous subsets of information deemed to be of interest, then the possibility of a disagreement must be considered. This is the problem of coordinating incomplete and possibly conflicting reports made by autonomous agents, with the purpose of reaching a decision on which action to take.

Rational decision-making is weakly defined in [S76b] as the process of choosing among a finite number of acts by a series of steps that

(i) lists the acts,
(ii) determines all their consequences, and
(iii) makes a comparative evaluation.

Although the definition is of little use as such, its weaknesses make it suitable for use as a proviso for some points made in this appendix. Note that the term *act* is loosely used, and the concept of *strategy* is used here instead. A more detailed discussion can be found in [L92]. A classical problem concerning (iii) is that there exists no absolute notion of rational decision-making. Rather, rationality is usually interpreted as meaning that any agent behaviour being sub-optimal with respect to the goal is either accidental or unavoidable. To explicate this interpretation, one may turn to the first presidential address of AAAI [N81] which has been influential in spreading the agent metaphor. Drawing upon ideas put forward by McCarthy in the late 1950s, Newell suggested his principle of rationality: "*If an agent has knowledge that one of its actions will lead to one of its goals, then the agent will select that action […] The principle of rationality provides, in effect, a general functional equation for knowledge. The problem for agents is to find systems at the symbol level that are solutions to this functional equation, and hence can serve as representations of knowledge* […] *The principle of rationality corresponds at the symbol level to the processes (and associated data structures) that attempt to carry out problem-solving to attain the agent's goals.*" [op. cit. pp.8–14].

The concept of rationality was initially treated in the MAS area as merely another property that agents could have, along with, e.g., autonomy, mobility, and benevolence (cf. Chapter 2 of [G86]). This development undoubtedly came about as a reaction to the view proposed earlier by traditional AI that cognitive capabilities are more important than an agent's means to communicate, react, or adapt. In the most extreme MAS frameworks, rationality is treated as an emergent feature of an agent system [B86].

The prime evaluation principle suggested is based on the principle of maximising the expected value (PMEV) since that principle is at the core of rational agent behaviour.[1] In the last few years, several researchers within DAI have equated rationality with the use of PMEV as a decision rule (see, e.g., [GD93]). However, this principle is not the only reasonable candidate for a decision rule. There are many reasons not to identify rationality with the PMEV, some of them well-known to game theorists [BE95]. The unrealistic assumption that the perfectly rational (or even hyper-rational, see [R92], p.107) players of the game have full knowledge of the game structure, and of the rationality of their opponents, is necessary to attain the desired equilibria [BC92]. Even if one accepts that game-theoretical decision rules cannot always provide useful advice to agents in non-ideal games, a view now seemingly assumed in computer science [R93], there remain difficult problems to face [M92].

As mentioned in Chapter 5, a number of other rules have been suggested by various researchers. One conclusion from that chapter is that it seems plausible to *supplement* a method based on PMEV with other rules. The strategies might be evaluated relative to a set of security levels considering how risky the strategies are. Moreover, it can be investigated in which parts of the hull those conditions are met. This is accomplished by using contractions for security levels as well.

Agent Modelling

In the agent model that underlies this approach, the DMA[2] faces a situation involving a choice between a finite set of strategies $\{S_i\}$ having access to a finite set of autonomous agents $\{A_i\}$ reporting their

[1] To be more precise, it should be called the principle of maximising the reported value. The credibilities represent importance weights given to individual reports. The aggregation should therefore be considered a weighted report value rather than an expectation.

[2] A DMA may be a human coordinator as well as another agent process.

opinions on the strategies to the DMA, see Figure A.1. Each of these agents may itself play the role of decision making agent, and the theory is independent of whether there is a specific coordinating agent or not. In other words, the focus in this appendix on a particular DMA is a matter of convenience. However, for the agents to carry out their tasks and to acquire sufficient and reliable knowledge en route, it is fundamental that they are able to evaluate information gathered from different sources, some unreliable and some noisy. The dynamic adaptation taking place over time as the agents interact with their environment, and with other agents, is affected by the means available to assess and evaluate imprecise information.

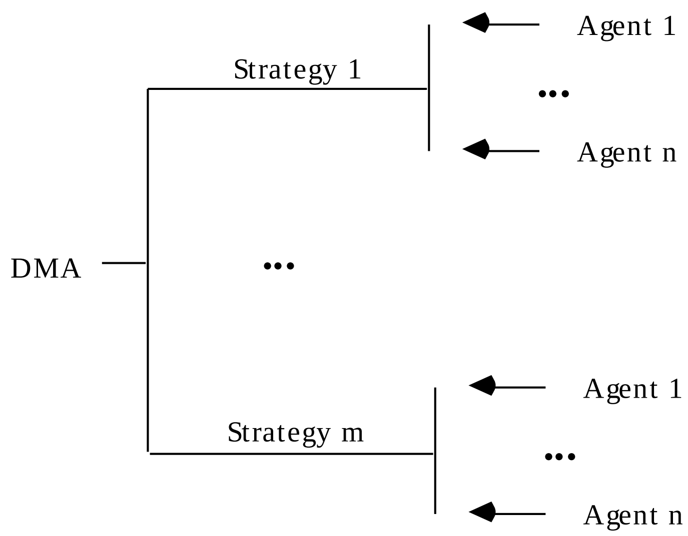

Figure A.1 A multi-agent decision model

In a situation modelled as in Figure A.1, some agents may be more reliable than others when evaluating the strategies involved, since different agents may have different capabilities to determine the values. The DMA may also have access to assessments expressing how trustworthy the different agents are. In the model, the DMA is set on choosing the most preferred strategy given the agents' individual reports and their relative credibility. The statements are assumed to be assigned and revised, typically with incomplete background information, and the evaluation method allows for vague and numerically imprecise information. Thus, the DMA may rank the credibilities of the different autonomous agents as well as quantify them in imprecise terms. The autonomous agents have a similar expressibility regarding their respective opinions about the strategies under consideration.

Example A.1: Assume a simplified scenario where a set consisting of the agents A_1, A_2, A_3, and A_4 report to a decision-making agent DMA on their respective opinions concerning the strategies for managing a system communications resource. The DMA has to decide whether to keep all time slots open for negotiation, to allocate some fixed bandwidth for high-volume users, or to lease out some of the bandwidth to neighbouring systems. Call these strategies S_1, S_2, and S_3, respectively. Further, assume that the agents A_1 through A_4 have reported to the DMA the following value statements.[3] The values involved could, for example, be monetary. In that case, they are linearly transformed to real values in the interval [0,1].

Statements according to agent A_1:

- The value of strategy S_1 is between 0.50 and 0.70.
- The value of strategy S_2 is between 0.10 and 0.70.
- The value of strategy S_3 is at least 0.30.

Statements according to agent A_2:

- The value of strategy S_1 is between 0.10 and 0.50.
- The value of strategy S_2 is between 0.40 and 0.70.
- I have no opinion about the value of strategy S_3.

Statements according to agent A_3:

- The value of strategy S_1 is not less than that of S_2.
- The value of strategy S_3 is between 0.50 and 0.70.

Statements according to agent A_4:

[3] The agents may have evaluated the prospective strategies using any number of well-established datacom traffic models. Here, only the evaluation of the total throughput situation is considered.

- The value of strategy S_2 is not less than that of S_3.
- The value of strategy S_1 is between 0.50 and 0.70.
- The value of strategy S_2 is at most 0.70.

Moreover, the DMA has estimated the credibility of A_1 through A_4 as numbers in the interval [0,1]. The number 0 denotes the lowest possible credibility, and 1 the highest:

- The credibility of agent A_1 is between 0.20 and 0.90.
- The credibility of agent A_2 is between 0.10 and 0.30.
- The credibility of agent A_3 is between 0.20 and 0.70.
- The credibility of agent A_4 is at most 0.50. ■

The rest of this appendix describes how the DMA may use the DELTA method in evaluating multi-agent problems such as the one above. A significant feature of the method is that it encourages the agents not to present report statements with an unrealistic degree of precision. Essentially, the model consists of a set of agents, a set of strategies, and two systems of statements concerning the credibilities and values involved. The sets of credibility statements and value reports are transformed into bases of linear constraints. The properties of those bases are discussed next.

Credibility Bases

A *credibility base* K with m agents is expressed in the credibility variables $\{c_1,\ldots,c_m\}$, stating the relative credibility of the different agents. The term c_k denotes the credibility assessment of agent A_k. A credibility base contains expressions about the credibility of each agent. To make the qualitative statements of credibility computable, they are translated in a manner similar to the standard DELTA method. Here, four types of possible credibility statements will be discussed. For a longer discussion of the parameters involved in the translations, refer to the corresponding treatment of probability statements in Chapter 4.

1. The credibility of A_k equals a number r, is at least r, is at most r.

   Example: The credibility of A_k is greater than r.

Translation: $c_k \in [r+\eta_1, r+\lambda_1]$

2. The credibility of A_k is between some real numbers.

   Example: The credibility of A_k is between r_1 and r_2.

   Translation: $c_k \in [r_1-\varepsilon_1, r_2+\varepsilon_1]$

3. The credibility of A_k is equal to the credibility of A_j, is approximately equal to that of A_j, is not less than that of A_j, etc.

   Example: The credibility of A_k is equal to the credibility of A_j.

   Translation: $c_k - c_j \in [-\varepsilon_2, \varepsilon_2]$

4. Agent A_k is credible, the opinion of agent A_k is worth considering, agent A_k is not credible, etc.

   Example: Agent A_k is credible.

   Translation: $c_k \in [r_3, r_4]$

In order for the credibility statements to be normalised, the constraint $\sum_k c_k = 1$ is added to the constraints above. The conjunction of constraints of the four types above, together with the normalisation, is the credibility base.

Example A.1 (cont'd): The DMA has estimated the credibility of A_1 through A_4 as numbers in the interval [0,1]. The translation of the statements into a credibility base results in the following expressions.

$c_1 \in [0.20, 0.90]$
$c_2 \in [0.10, 0.30]$
$c_3 \in [0.20, 0.70]$
$c_4 \in [0.00, 0.50]$

The credibilities are subject to the normalisation constraint $\sum_k c_k = 1$. Consequently, the greatest value that can consistently be assigned to c_1 is 0.7 (the minimum value that $c_2 + c_3 + c_4$ can have is 0.3, since $c_1 + c_2 + c_3 + c_4$ should be 1). Since no other weight is affected, the hull of this base is $\{\langle 0.20, 0.70\rangle, \langle 0.10, 0.30\rangle, \langle 0.20, 0.70\rangle, \langle 0.00, 0.50\rangle\}$. ■

Report Bases

A report base R contains statements about individual agents' opinions of the values of different strategies, i.e., it consists of a number of interval constraints and core intervals that represent various strategy statements. It is expressed in value variables $\{v_{11},\ldots,v_{1n},\ldots,v_{m1},\ldots,v_{mn}\}$ stating the values of the strategies according to the different agents. The term v_{ik} denotes the value of strategy S_i according to the report of agent A_k. Five types of possible report statements are handled.
Given an autonomous agent A_k:

1. The value of the strategy S_i equals *r*, is at least *r*, etc.
   Example: The value of S_i is greater than r.
   Translation: $v_{ik} \in [r+\eta_1, r+\lambda_1]$
2. The value of strategy S_i is between some real numbers.
   Example: The value of S_i is between r_1 and r_2.
   Translation: $v_{ik} \in [r_1-\varepsilon_1, r_2+\varepsilon_1]$
3. The strategy S_i is as desirable (or undesirable) as strategy S_k, more desirable than S_k, the value of S_i is approximately equal to the value of S_k.
   Example: The strategy S_i is as desirable as S_j.
   Translation: $v_{ik} - v_{jk} \in [-\varepsilon_2, \varepsilon_2]$
4. The difference in value between S_i and S_j is not less than the difference in value between S_m and S_n.[4]
   Translation: $(v_{ik} - v_{jk}) - (v_{mk} - v_{nk}) \in [\eta_1, \lambda_1]$
5. The strategy S_i is desirable, S_i is fairly desirable, S_i is undesirable, etc.
   Example: The strategy S_i is desirable.
   Translation: $v_{ik} \in [r_3, r_4]$

Example A.1 (cont'd): The reports provided by the agents are translated into the following expressions.[5]

$v_{11} \in [0.50, 0.70]$ $v_{33} \in [0.50, 0.70]$

[4] For simplicity, it is assumed that the value of S_i is greater than the value of S_j, and that the value of S_m is greater than the value of S_n.

[5] The constants in the translations are chosen to keep the presentation simple.

$v_{21} \in [0.10, 0.70]$ $v_{14} \in [0.50, 0.70]$
$v_{31} \geq 0.30$ $v_{24} \leq 0.70$
$v_{12} \in [0.10, 0.50]$ $v_{13} \geq v_{23}$
$v_{22} \in [0.40, 0.70]$ $v_{24} \geq v_{34}$

This report base is then subject to evaluations using aggregate rules or security levels. ■

Agent Decision Frames

A credibility base K together with a report base R constitute an *agent decision frame* ⟨S,K,R⟩, where S is the set of strategies. This is in analogy to the ordinary decision frame ⟨C,P,V⟩ in the standard DELTA method. The mapping onto ordinary frames is straightforward. The strategies correspond to consequence sets, and the report elements are analogous to the consequences. Further, the credibilities have properties similar to probabilities, and report values are almost the same as values in the ordinary frame.

The mapping is not perfect, though. At first, it seems that credibilities map directly onto probabilities in that they have a similar role, distributing mass over the report values. But if credibilities are allowed to be assigned per strategy for each agent, then a more credible report about v_{ik} from the agent A_k might be forced to assume a lower credibility than a less credible report about v_{jk} from the same agent due to other agents also being more credible when giving reports about strategy S_i and the credibilities being normalised to sum to one.[6] Thus, only one credibility assessment per agent ought to be allowed. Still, since it is a normalised mass to be distributed, it might be more reasonable to interpret credibilities as weights instead. If there are no credible reports, the agents' credibilities must sum to one, and conversely, if all reports are very credible, they must still sum to one. This is not in accordance with the common interpretation of credibility. Finally, if an agent A_i has

[6] If there would be no normalisation, then the aggregated value would not make sense.

low credibility and another agent A_j has a much higher credibility, the statement $v_{ik} > v_{jk}$ has the same effect regardless. These discrepancies must be accounted for in an agent decision model. Such problems notwithstanding, the DELTA method is well-suited for multi-agent systems.

Comparing Strategies

Relative to a particular agent decision frame, which strategy should be chosen? The problem formulation is mathematically almost equivalent to the decision frame in Chapters 4–6, thus rendering the method and computational machinery of those chapters suitable for this task as well. As is the case for ordinary decision frames, for agent frames it is often not enough to determine the set of non-dominated (admissible) strategies, since in non-trivial decision situations this set is too large, i.e. the admissible strategies are too numerous and the DMA cannot adequately discriminate between them. Moreover, when approaching a problem, the autonomous agents as well as the DMA are encouraged to be deliberately imprecise, and thus values close to the boundaries of the interval constraints seem to be the least reliable ones. This is a typical case for applying the contraction principle as described in Chapters 4–5, and in the example, the effects of contraction can be seen. Note that no core is specified, and the contraction goes from the hull inwards to the degree of 80%.

Example A.1 (cont'd): Entering the information into DELTA results in the agent decision frame in Table A.1.

```
Frame 'ExA1' in folder 'PhD' has 3 strategies
S1 (Strategy 1)
S2 (Strategy 2)
S3 (Strategy 3)

Each strategy is valued by 4 agents
A1 (Agent 1)
A2 (Agent 2)
A3 (Agent 3)
A4 (Agent 4)

The credibility base contains 4 constraints
 1: C1 ∈ [0.200,0.900]
```

```
 2: C2 ∈ [0.100,0.300]
 3: C3 ∈ [0.200,0.700]
 4: C4 ∈ [0.000,0.500]

Credibility hull    Symmetry hull
C1 ∈ [0.200,0.700]  [0.200,0.494]
C2 ∈ [0.100,0.300]  [0.100,0.218]
C3 ∈ [0.200,0.700]  [0.200,0.494]
C4 ∈ [0.000,0.500]  [0.000,0.294]

The report base contains 10 constraints
 1: V1.1 ∈ [0.500,0.700]
 2: V2.1 ∈ [0.100,0.700]
 3: V3.1 ∈ [0.300,1.000]
 4: V1.2 ∈ [0.100,0.500]
 5: V2.2 ∈ [0.400,0.700]
 6: V3.3 ∈ [0.500,0.700]
 7: V1.4 ∈ [0.500,0.700]
 8: V2.4 ∈ [0.000,0.700]
 9: V1.3 - V2.3 ∈ [0.000,1.000]
10: V2.4 - V3.4 ∈ [0.000,1.000]

Report hull
V1.1 ∈ [0.500,0.700]
V1.2 ∈ [0.100,0.500]
V1.3 ∈ [0.000,1.000]
V1.4 ∈ [0.500,0.700]
V2.1 ∈ [0.100,0.700]
V2.2 ∈ [0.400,0.700]
V2.3 ∈ [0.000,1.000]
V2.4 ∈ [0.000,0.700]
V3.1 ∈ [0.300,1.000]
V3.2 ∈ [0.000,1.000]
V3.3 ∈ [0.500,0.700]
V3.4 ∈ [0.000,0.700]

Focal point
Cred:  0.347  0.159  0.347  0.147
Agent     A1     A2     A3     A4
S1:    0.600  0.300  0.500  0.600
S2:    0.400  0.550  0.500  0.350
S3:    0.650  0.500  0.600  0.350
```

Table A.1 The agent decision frame

Evaluating the frame above results in Tables A.2–A.4.[7] Table A.2 shows the contraction of strategy S_1.

```
Contraction      0%     20%     40%     60%     80%
S1      min:  0.166   0.247   0.322   0.392   0.458
```

[7] The output from DELTALIB (see Chapter 3) is numeric. DMAs, especially software agents, often desire to receive the evaluation results in the form of matrices or tables instead of graphs in order to perform numerical computations on them.

	mid:	0.518	0.518	0.518	0.518	0.518
	max:	0.828	0.758	0.691	0.629	0.572

Table A.2 The contraction of S_1

Tables A.3–A.4 show the contractions of the strategies S_2 and S_3, respectively. Hence, strategy S_2 is inferior to both S_1 and S_3, but strategy S_3 is slightly better than S_1. A further investigation is recommended in order to identify critical variables.

Contraction		0%	20%	40%	60%	80%
S2	min:	0.060	0.143	0.223	0.300	0.375
	mid:	0.451	0.451	0.451	0.451	0.451
	max:	0.848	0.767	0.686	0.607	0.528

Table A.3 The contraction of S_2

Contraction		0%	20%	40%	60%	80%
S3	min:	0.186	0.269	0.349	0.424	0.496
	mid:	0.565	0.565	0.565	0.565	0.565
	max:	0.914	0.840	0.768	0.699	0.631

Table A.4 The contraction of S_3

It is natural to ask how sensitive the different contractions are to changes in the agent frame. The DMA can simultaneously vary any number of intervals to discover credibility or value variables that are especially critical. Assume that the DMA wants to investigate whether it is meaningful to allocate resources to agent A_1 for collecting additional information about strategy S_3. Before doing that, the DMA can investigate how influential the report from the agent would be. For instance, the DMA can restrict the maximum value of v_{31} to 0.6 instead of 1 and evaluate the modified decision situation. Table A.5 shows the result for strategy S_3. The strategy is now slightly worse than S_1. The new information does not change the results in Tables A.2 or A.3.

Contraction		0%	20%	40%	60%	80%
S3	min:	0.186	0.257	0.324	0.386	0.443
	mid:	0.495	0.495	0.495	0.495	0.495
	max:	0.745	0.695	0.645	0.596	0.546

Table A.5 The result of modifying S_3

Thus, it is reasonable to allocate resources to collect more information about strategy S_3 from agent A_1. The DMA may now interactively proceed in this way to investigate critical reports in order to gain a better understanding of the decision problem and finally reach a conclusion. ■

Security Levels

The intuition behind security levels is that they provide limits beyond which a strategy is undesirable. Thus, a DMA might regard a strategy as undesirable if it has access to a report in which a credible agent assigns a low value to the strategy.

Example A.2: Suppose that the DMA has stipulated that a strategy S_i is undesirable **iff**

- according to some agent A_j, the value of strategy S_i is less than 0.45
- the credibility of that agent A_j is greater than 0.65.

Assume that v_{12} is in the interval [0.40, 0.60] and that c_2 is in the interval [0.20, 0.70]. Then S_1 is below the threshold and is thus undesirable. It is advisable to investigate how much the different intervals can be decreased while the security levels are still violated. In this manner, the stability of the result can be studied. For example, it can be seen that strategy S_1 ceases to be undesirable when the left end-point of the interval of v_{12} is increased by 0.05. Consequently, the result is quite unstable. ■

The example contained a very simplistic approach to limiting undesirable outcomes. To be more sophisticated and utilise the DELTA method, the concept of security level as defined in Chapter 5 is applied. There, an observation regarding security levels was made, which is here turned into a definition of agent security levels and is put to use in testing which strategies might be undesirable.

Definition A.1: Given an agent decision frame ⟨S,K,R⟩ and two real numbers r,s ∈ [0,1], a strategy S_j *violates agent security level s for value threshold r* **iff** for K_j = {k ™ v_{jk} ≥ r} $\sum_{k \in K_j} c_k \leq 1 - s$.

This is best illustrated by an example which evaluates the security levels using weak first order dominance.[8]

Example A.1 (cont'd): Using the definitions above, it may now be investigated to what extent the different strategies are undesirable. Figure A.2 shows, for each strategy and a value threshold of 0.10, the worst possible credibility assignments consistent with the frame for different degrees of contraction, i.e. the security levels violated by weak dominance. In the figure, the K- and R-bases are contracted simultaneously, but this is not the only option. The K-base might be left uncontracted, studying only the R-base under contraction, and conversely, the R-base might be untouched while contracting K.

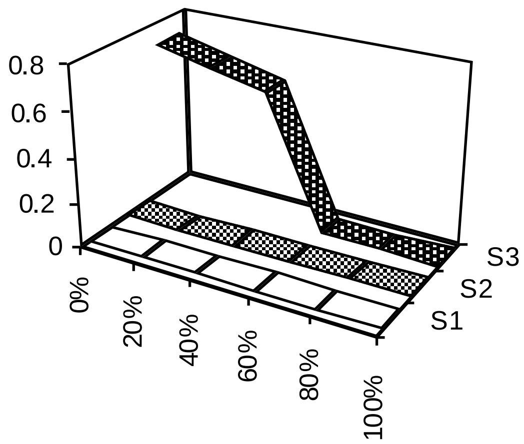

Figure A.2 Value threshold 0.10

From the figure, it can be seen that the strategies S_1 and S_2 are not undesirable in any part of the decision frame. Strategy S_3 is undesirable in the original frame and remains so until it is contracted by more than 60%. For instance, when the decision frame is con-

[8] See Chapter 5 for an explanation of weak dominance.

tracted by 40%, the greatest joint credibility for the bad reports of this strategy is 0.58.

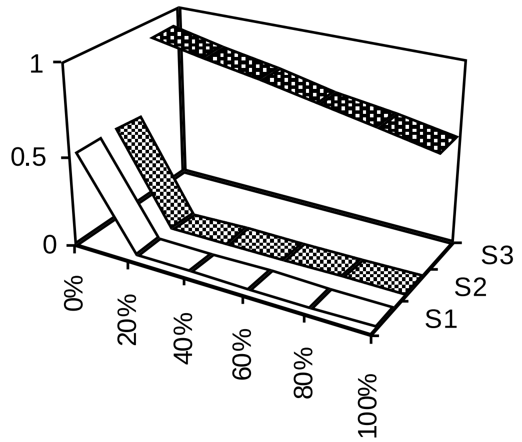

Figure A.3 Value threshold 0.20

Figures A.3 and A.4 show the evaluations for the value thresholds 0.20 and 0.50 respectively. As can be seen in Figure A.3, the strategies S_1 and S_2 are now undesirable in some parts of the decision frame. However, they cease to be undesirable at contractions of at least 20%.

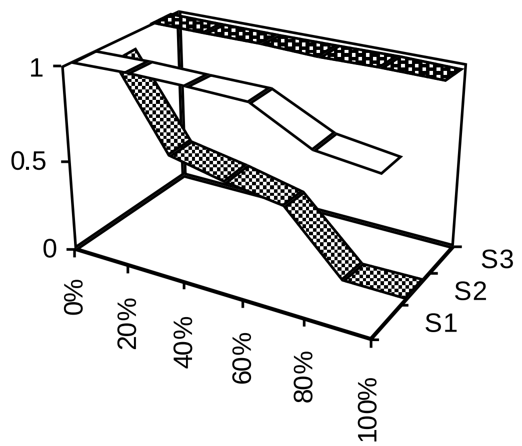

Figure A.4 Value threshold 0.50

Figure A.4 shows that for very high value thresholds, S_3 is undesirable regardless of the degree of contraction. Thus, it can be seen that the results of the evaluation are strongly dependent on boundary values, and consequently they should be further investigated in sensitivity analyses. While S_1 and S_3 were preferable to S_2 already in the primary evaluation above, S_3 seems to be too dangerous to adopt as the main strategy for the time slot allocation. Thus, the agent selects strategy S_1 – keeping all the time slots open for negotiation. ■

By using security levels, the decisions made by the DMA will be more reliable and predictable than if such levels were not imposed on the reports. The trust the DMA can put in the results will increase considerably as it is able to set the levels and thresholds according to its appreciation of the particular decision problem.

Long you live and high you fly
Smiles you give and tears you cry
All you touch and all you see
Is all your life will ever be

Run, rabbit run
Dig that hole, forget the sun
And when at last the work is done
Don't sit down, it's time to dig another one

For long you live and high you fly
But only if you ride the tide
And balance on the biggest wave
You race towards an early grave

– R. Waters

B

Risk Management

This appendix deals with another application area of computational decision analysis methods, the area of risk management. The content of the appendix is joint work with Love Ekenberg, IIASA, and Anders Elgemyr, ROA. The text is partly derived from [ED95] and [DEE96].

The risk analysis method DEEP (Damage Evaluation and Effective Prevention) substantially extends the evaluative phases compared with earlier approaches. The concept of risk analysis is used in a little wider sense than usual. Often, only identification and valuation of damage risks are included in the concept, but here selection of risk treatment, risk financing, and analysis of the measures taken are also included. The presentation is focused on the identification and analysis of threats and on the evaluation of the suggested actions since those are the steps where the DEEP method differs the most from other methods. The other steps are fairly well covered in other texts.[1] The idea behind DEEP is to offer an analytical framework for enhancement of the chain identification–valuation–action in risk management without aiming at replacing it.

[1] Risk analysis is less general in its first steps. In different industries, the values to be protected and the threats are fairly industry specific. It is therefore not surprising that, for example, the chemical industries in Sweden publish a text applicable specifically to their own needs [K96]. But also the later evaluation steps are treated as if they were industry specific. This might be due to the lack of general methods that seem to fit in different industries, see for example [EM92].

To acquire a satisfactory understanding of the risk situation, management often desires some kind of structured approach to the analysis. Thus a risk analyst, conducting a risk analysis, frequently has access to standard procedures for identifying and assessing threats and for identifying and valuating assets. A tentative list of basic steps in risk management could be the following:

- Identify the assets/objects that should be protected.
- Identify the threats that should be protected against.
- Estimate the probabilities for the threats to materialise.
- Estimate the values lost if the threats materialise.
- Assess the current protection.
- Decide which threats to rectify and which to leave unmanaged.
- Evaluate which protective measures are reasonable to take.
- Find financing for a reasonable part of the remaining risk.
- Execute the decided plans.
- Follow up on the effectiveness and efficiency of the plans.

In the analysis, different threats are compared to each other, and those not found to be serious are filtered out. The others are ranked in order of treatments necessary. Below, some risk models are criticised for not being able to rank the seriousness of different threats. In the evaluation step, the possible courses of action are specified. Although in real life such analyses are often carried out, this step is left out in most existing risk analysis models. This is a clear deficiency that may substantially reduce the applicability of analysis results.

For insurance management problems, for example, different problems are encountered depending on the type of insurance. For high-volume, high-frequency incidents, insurance companies have a well-developed set of mathematical and statistical tools at their disposal when calculating the cost of insurance. The risk management issue is to keep such insurances or not, balancing the decision against the profit margin for the insurance companies and assuming a reasonably well-working insurance market with at least rudimentary competition mechanisms. For low-frequency risks, the situation quite is different. Insurance statistics is not as good a tool, but the need for risk analysts to have tools at their disposal is perhaps even greater. This poses some hard challenges to risk staff in general and to risk managers in particular.

To make it easier to grasp the ideas behind the DEEP method, to compare it with traditional approaches, and to indicate some of their disadvantages, a brief survey of some approaches to risk analysis is included. Ensuing this, an informal overview of the method is given, followed by a description of its evaluation step incorporating DELTA.

Risk Evaluation Approaches

Different decision methods are used for assessment in risk analysis. They are typically involved in several steps to identify and evaluate assets, such as properties and information, and to identify and evaluate threats, such as fire, burglary, and industrial espionage. Such analyses are also carried out to verify the current protection, and to evaluate the effects of modifying it.

Often, when evaluating the cost of an incident, the model requires numerically precise data. A main problem is that in real-life analysis it is often impossible for an analyst to explain the difference between closely proximate probabilities, for example 23% and 25%. The problem is emphasised by the inability to express varying reliabilities for different pieces of information. Which data are based on long experience, and which are mere guesses? In models using numerically precise information, this kind of expressibility is severely limited.[2] The following three sub-sections focus on two common techniques used in risk evaluations and a more powerful approach, the expected cost.

[2] Methods for estimating the monetary cost of a simple incident by using numerically precise data in an expected cost model can be found in, e.g., [D90, pp.86 ff.].

Point Scale Models

One attempt to overcome the unrealistic and time-wasting assumption of numerically precise information is to be more imprecise, even in making the estimates. Broder writes: "*[…] it is neither necessary nor desirable to make precise statements of impact and probability. The time needed for the analysis will be considerably reduced and its usefulness will not be decreased if impact (i) and frequency (f) correlations are given in factors of 10.*" [B84, p.22]. Then he proposes the following scale:[3]

Loss valuation of an incident		**Estimated frequency to occur**	
$10	i = 1	Once in 300 years	f = 1
$100	i = 2	Once in 30 years	f = 2
$1,000	i = 3	Once in 3 years	f = 3
$10,000	i = 4	Once in 100 days	f = 4
$100,000	i = 5	Once in 10 days	f = 5
$1,000,000	i = 6	Once per day	f = 6
$10,000,000	i = 7	10 times per day	f = 7
$100,000,000	i = 8	100 times per day	f = 8

Table B.1 Broder's point scale

The annualised loss expectancy is then approximated by $\frac{10^{(f+i-3)}}{3}$.

A problem with this approach is that the possible values and frequencies are spaced too far apart. This can be solved by using decimal numbers for *i* and *f*, but then the reasoning is back where it began. Furthermore, an important feature of a method allowing imprecise data should be enabling the detection of critical variables and the study of what effects modifications to the given data will have. This is not least important when the possible values are spaced far apart. Also, a risk analyst using point scales is still unable to express varying degrees of reliability for the different pieces of information.

[3] The method was originally suggested in [C77] and is recommended to prospective U.S. government suppliers by NIST.

Risk Level Models

One way to partially overcome the problems with point scale models is to allow the analyst to express the different values in non-monetary terms. In Sweden, for instance, a relative three-level model has been used for example by [H88, SAF86, W91b]. The probabilities and values involved (somewhat misleadingly called consequences in this approach) are expressed as shown in Figure B.1. Variants of the three-level model are also frequently used. For example, [S89–91] uses a four-level model, as does the Swedish SBA method [W84]. Not infrequently, even more rudimentary models are proposed.[4]

	Probability	Value	Risk level
1	**Low/Small** Seldom occurs	**Small** Low cost, little damage or loss	**Acceptable** Can be allowed Should be remedied
2	**Medium** Occurs neither often nor seldom	**Medium** Greater cost Greater damage or loss	**Unacceptable** Not allowed Must be remedied
3	**Great/High** Often occuring	**Great** Cost cannot be borne Total loss	**Catastrophic** Must be remedied immediately Unforgivable

Figure B.1 From [H88, p.76].

The *risk level* is a function of the sum (not product) PV = *probability* + *value*. If PV ∈ {2}, the risk level is 1, if PV ∈ {3,4}, the risk level is 2, and if PV ∈ {5,6}, the risk level is 3. A major problem with this approach is that the categories are too wide, with no discrimination within them. Therefore, most risks evaluate to risk level 2 with no indication of

[4] Many practitioners have abandoned the concept of probability altogether. For instance, insurance advisors often find it too hard to make estimates of the frequencies of accidents because of low levels of repetition, and they sometimes erroneously draw the conclusion that all kinds of probability based reasoning should be avoided. For example, in [G92b] a five-level model without probabilities is suggested and in [ESF91] probabilities are also ignored.

how to order the risks within that level. A competent risk analyst is capable of differentiating between disastrous, unacceptable, and acceptable risks without the aid of decision tools. The problem is to decide the order and the extent of the reduction needs of different unacceptable risks. Hence, when the risk situation is obvious, there is little need for a model, and when it is not, the models offer little help.

Expected Cost Models

The choice of the formula above for evaluation seems peculiar, and it is obvious that what results from it differs from evaluations using the expected value, which can be formulated in risk analysis terms as follows. The first definition covers the costs of actions, and below costs of incidents are defined as well. They differ conceptually as in the former the probabilities refer to possible incidents following actions, while in the latter the probabilities refer to possible effects of an incident. Example B.2 below uses expected cost in the first sense.

> **Definition B.1**: An *action* A_i may result in a number of possible incidents $\{H_{i1},\ldots,H_{in}\}$. The *expected cost of an action* A_i can be expressed as $p_{i1}\cdot c_{i1} +\ldots+ p_{in}\cdot c_{in}$, where c_{ik} denotes the cost of the incident H_{ik}, and p_{ik} denotes the probability of H_{ik} occurring given that action A_i is taken.

In a corresponding way, the definitions can be expressed in terms of incident costs instead and the expected cost should be minimised. When analysing the consequences of an incident, not only monetary costs are of interest. Thus, the concept of cost will be used in a more general sense, including both quantitative and qualitative values. Utilities could have been used instead, but in this context, cost is a more natural concept than utility. Note that monetary cost is a special case of cost.

The first pure risk concept to be considered is *simple incidents* (resulting in only direct consequences), which then will be extended to *incidents* (resulting in both direct consequences and new incidents).

> **Definition B.2**: A *simple incident* H_i has a number of possible consequences $\{C_{i1},\ldots,C_{in}\}$. The *expected cost of a simple incident* H_i can be expressed as $p_{i1}\cdot c_{i1} + \ldots + p_{in}\cdot c_{in}$, where c_{ik} denotes the cost of the consequence C_{ik}, and p_{ik} denotes the probability of C_{ik} occurring given that the incident H_i occurs.

It is possible to generalise the description of a simple incident resulting in a set of consequences. The new description allows an incident to generate both new incidents *and* consequences, which in turn can generate even more incidents and consequences, see Figure B.2. The H's in the figure denote incidents, and the C's different consequences. The P's denote the probabilities involved.

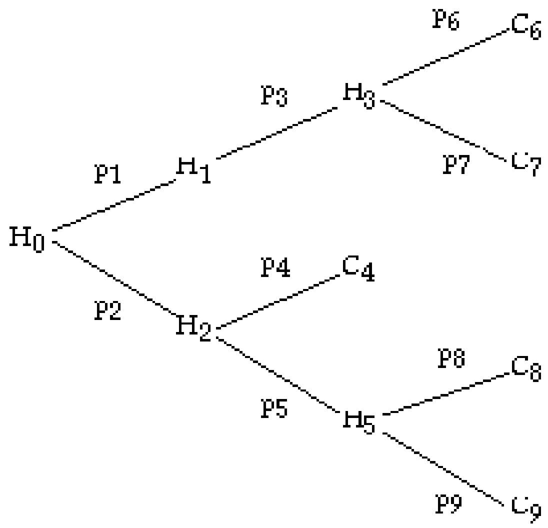

Figure B.2 An extended consequence analysis

Now, the definition of expected cost is extended. Note that in the following definitions, an incident is formally a set of consequences and incidents.

> **Definition B.3**: A set of incidents and simple incidents $\{H_1,\ldots,H_r\}$ is an *incident*. The *expected cost of an incident* $\{H_1,\ldots,H_r\}$ is expressed by the formula $E_i = p_{i1}\cdot E_1 + \ldots + p_{ir}\cdot E_r$, where E_k denotes the expected cost of the incident (or simple incident) H_k, and p_{ik} denotes the probability of the incident H_k given H_i.

Example B.1: Consider Figure B.2.[5] The incident H_5 can result in C_8 and C_9, and only these. Hence, H_5 is a simple incident, and the expected cost of it is equal to $p_8 \cdot E_8 + p_9 \cdot E_9$. The incident H_2 generates a new incident H_5 and can also result in C_4. The expected cost of the incident H_2 is therefore equal to $p_4 \cdot E_4 + p_5 \cdot E_5$. E_4 is the cost of the simple incident consisting of the single consequence C_4, and E_5 (= $p_8 \cdot E_8 + p_9 \cdot E_9$) is the expected cost of the simple incident H_5. ■

The discussion about evaluation below is based on a one-level description, i.e., an incident does not generate new incidents. This does not cause any real restriction, because as mentioned in Chapter 1, a multi-level tree problem (where an incident generates new incidents) can always be transformed into a one-level problem. Before the evaluation, the next section presents the method in general.

The DEEP Method

This section describes the DEEP method and how it may be used to evaluate the effects of different actions to prevent possible incidents. By using the method, it is easier to realise which threats are the most important to handle and what effects will follow from the treatments. It is also important that the method can be adjusted to the risk policies of the specific companies using it.

Nine Risk Analysis Steps

The DEEP method is a systematic model for risk analysis using sophisticated methods for calculating in which order different threats should be handled as well as comparing different actions to each other. The analysis method is divided into nine steps.

An overview of the process is pictured in Figure B.3. The numbers in the figure relate to the steps in DEEP.

[5] For clarity, the indices have been simplified in the example.

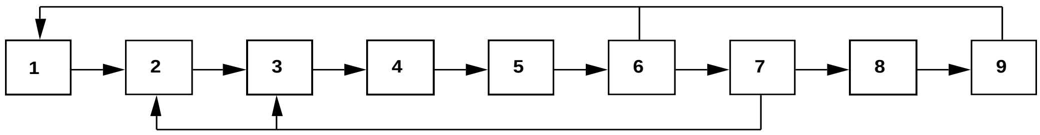

Figure B.3 The DEEP steps

The nine steps follow naturally after each other and comprise everything from investigating possible incidents to sensitivity analysis of the risk analysis. In every step, the results are documented in order to be able to easily return for a renewed analysis should the preconditions for the original analysis have been partially changed. Steps 1–3 and 8–9 are discussed only superficially, as this part of the thesis deals with applications of computational decision analysis and not risk analysis per se. The first three steps aim at providing a picture of the current risk exposure of the organisation under analysis.

1. Scope Analysis

When a risk analysis is planned, it is important to state clear goals for the analysis and delimit its scope. Seldom an entire corporation is to be analysed at the same time, and *Step 1* includes dividing the analysis into suitable parts and risk areas. A decision is often made only to handle pure losses, incidents that only generate costs since then it is easier to apply rational decision processes.[6]

2. Possible Damage

The *second step* in DEEP is to closer examine those parts of the company or organisation that are included in the analysis. Which incidents may occur? Which other incidents may follow as a result of primary damage? To what extent will the production process be interrupted? It is important to systematically identify all potential objects in danger of being damaged and all events that lead to damage to property,

[6] The other option would be to include risks that could result in incomes as well, so called business risks.

personnel, process interruption, liabilities, etc – not only the results of an incident.

3. Current Protection

Ensuing that, it is natural to closely study the current protection. It consists of both direct protection and indirect protection in the form of insurance. Typical questions in *Step 3* include: Is the protection level sufficient? What happens if the protection devices do not work as expected? Which is the appropriate balance between direct protection and insurance? The third step is concluded by investigating possible treatments. For every possible incident that has been identified, some alternative protections are listed. They should be at least two – keeping the current protection and improving it in some way. Often, there is more than one way of reducing the risk, and those alternatives differ with respect to costs and effects. For example, spreading the risk can be done in several ways, physically by changing the flow of work and goods or monetarily by increasing the level of insurance. Another example is reducing the risk, either by pre-incident actions (which decrease the probability of an incident occurring) or by post-incident actions (which decrease the cost of an incident that has already occurred).

4. Probabilities

The next two steps contain statements of probabilities and costs. For all alternative actions, the probabilities for the possible incident and the cost (or value) for the damage given that action are stated. This is done relative to the list of possible actions from the previous step. *Step 4* contains estimates of probability. To perform a reasonable risk analysis, it is necessary to estimate the frequencies of possible incidents. Sometimes, the frequency data available is sufficient, but in many cases the analyst must rely on more or less well-founded estimates.

5. Costs

In the same manner, *Step 5* contains the estimation of costs. This includes protection costs as well as costs incurred from damages. The costs can be expressed directly in monetary values or in some other appropriate scale. In those two steps, it is not unusual to find that the information available is insufficient and a supplementary investigation has to be made in order to achieve reasonable results. In these steps, it may even turn out that the problem has been structured in an unsuitable way, and that the terms of reference for the analysis have to be revised.

6. Evaluation

When all incidents have been identified and valued, it is time in *Step 6* to evaluate the alternative actions. Such an evaluation can be made with respect to different principles, for example minimising the expected loss. An important feature of the evaluation step is the ability to exclude acceptable risks from further evaluation with the aid of threshold levels.[7] If the potential cost for a specific risk is below the policy level of top management, it may be classified as acceptable and no more resources need to be used for further analysis of the accompanying threats.

7. Sensitivity Analysis

Even a thorough analysis may have much to gain from being subject to a sensitivity analysis, which is the purpose of *Step 7*. In this step, the probabilities and costs are altered in order to study the stability of the results. When the numbers are altered, the evaluation result will possibly change as well. Exactly where this occurs is interesting, because it indicates which input data is critical to the conclusions drawn. Those should be studied more closely since they help indicate the better use of the resources for analysis.

---

[7] Security levels through thresholds are described in Chapter 5 and Appendix A. Here, good alternatives are removed, but the reasoning involved is the same.

8. Implementation

When the evaluation process is concluded, the chosen actions are implemented in *Step 8*. This step is specific to the particular organisation and it also includes the financing of risks remaining after the actions have been taken. This financing could be done by using insurances.

9. Follow-up

After some time has elapsed, it is important to verify the results of the actions. Otherwise, the actions may have resulted in the problems being transferred to other problem areas, and *Step 9* is supposed to discover such problems.

As was explained above, during the analysis it may turn out to be necessary to collect further information or renew discussions made earlier. This feedback is illustrated by backward pointing arrows in the process in Figure B.3.

Evaluation in DEEP

When evaluating information from a consequence analysis, risk analysts using DEEP may use a formula expressing the expected cost of an incident, and this section demonstrates how the DELTA method can be modified to evaluate the expected cost in the same manner as the expected value is handled in Chapters 4–6.

A set of simple incidents is treated simultaneously since much can be gained from studying several interrelated incidents at the same time. The representation of *probabilities* is not considered here, since it is the same as in the original DELTA method of Chapter 4. The representation of *costs* is considered instead, the interpretations of admissible statements are formalised, and this is described for four types of possible cost statements.

1. The cost of the incident H_{ij} equals *m*, is at least *m*.

   Example: The cost of H_{ij} is greater than m.

Translation: $c_{ij} \in [m+\eta_1, m+\lambda_1]$

2. The cost of the incident H_{ij} is between some real numbers.

   Example: The cost of H_{ij} is between k_1 and k_2.
   Translation: $c_{ij} \in [k_1-\varepsilon_1, k_2+\varepsilon_1]$

3. The incident H_{ij} is as expensive as incident H_{ik}, more expensive than incident H_{ik}, the cost of incident H_{ij} is approximately equal to the cost of incident H_{ik}.

   Example: The incident H_{ij} is as expensive as incident H_{ik}.
   Translation: $c_{ij} - c_{ik} \in [-\varepsilon_2, +\varepsilon_2]$

4. The difference in cost between H_{ij} and H_{ik} is not less than the difference in cost between H_{im} and H_{in}.[8]
   Translation: $(c_{ij} - c_{ik}) - (c_{im} - c_{in}) \in [m+\eta_1, m+\lambda_1]$

The important point is that statements as above are translated into a system of linear inequalities that make them easy to handle in the DELTA method. If a risk analyst still is averse to the use of qualitative statements, he may use only interval statements instead.

The conjunction of expressions of the four types above is called the *cost base* K. The probability base and the cost base are linear systems and together constitute the *risk frame* ⟨C,P,K⟩. Evaluating a risk frame is mathematically equivalent to the evaluation of decision frames in Chapters 5–6. Hence, this appendix will not discuss those procedures but rather conclude with an example to illustrate the method.

Evaluation Example

The following example is supposed to show how the DEEP method works in steps 4–7. The much simplified numerical example concerns one burglary event during a given period and the estimates are imprecise. The purpose is to illustrate that the method can facilitate an

[8] For simplicity, assume that the cost of H_{ij} is greater than the cost of H_{ik} and that the cost of H_{im} is greater than the cost of H_{in}.

assessment as to which protective measures are reasonable even though only imprecise information is available.

Example B.2: A company desires to decrease its exposure to risk by installing more protective equipment and mechanisms at a certain production facility. The tax deduction period for such equipment is five years, and thus the analysis below is based on estimates of probability for a five year period.

First, the possible damages for the period are assessed. The assessment results in the following possible incident list.

H_1	No burglary attempts
H_2	All burglary attempts fail
H_3	A burglary succeeds

Table B.2 Incident list

The existing protective equipment is assessed and possible actions are listed. This list contains three possible alternative acts.

A_1	Keep the current protection
A_2	Add the improvements recommended by the insurance company
A_3	Additionally install more functionality as recommended by an independent security consultant

Table B.3 Action list

After that, an analysis commences which gives the following coarse estimates for the probabilities and costs for possible damages with respect to the different available courses of action. The costs listed include purchase costs for the equipment and costs for events that occurred.

Probabilities	No attempts	All attempts fail	Burglary
A_1 – Current protection	20–50%	10–20 %	30–60 %
A_2 – Insurance company	30–50%	20–50 %	15–30 %
A_3 – Ins.comp. + consultant	35–55%	30–60 %	10–20 %
Costs ($ million)	No attempts	All attempts fail	Burglary
A_1 – Current protection	0	0.1–0.3	2.5–6.5
A_2 – Insurance company	0.6–0.8	0.8–1.2	3.3–7.5
A_3 – Ins.comp. + consultant	2.2–2.6	2.4–3.1	5.2–9.1

Other statements

- The probability of 'No attempts' increases the more powerful protection is installed.
- The difference in costs between 'No attempts' and 'All attempts fail' is small if A_2 is chosen. It is estimated to be about \$0.2 to 0.4 million and is due to equipment only.
- Also the difference in costs between 'No attempts' and 'All attempts fail' is small if A_3 is chosen. It is estimated to be about \$0.2 to 0.5 million.

Table B.4 Estimated probabilities and costs

In this example, there are three incidents (H_1–H_3) to each of the three courses of action – the two additional protections plus keeping the current protection level during the period.

$p_{11} \in [20\%,50\%]$	$c_{11} \in [0.00,0.00]$	$p_{11} < p_{21} < p_{31}$
$p_{12} \in [10\%,20\%]$	$c_{12} \in [0.01,0.03]$	$p_{12} < p_{22} < p_{32}$
$p_{13} \in [30\%,60\%]$	$c_{13} \in [0.25,0.65]$	$c_{22} - c_{21} \in [0.02,0.04]$
$p_{21} \in [30\%,50\%]$	$c_{21} \in [0.06,0.08]$	$c_{32} - c_{31} \in [0.02,0.05]$
$p_{22} \in [20\%,50\%]$	$c_{22} \in [0.08,0.12]$	
$p_{23} \in [15\%,30\%]$	$c_{23} \in [0.33,0.75]$	
$p_{31} \in [35\%,55\%]$	$c_{31} \in [0.22,0.26]$	
$p_{32} \in [30\%,60\%]$	$c_{32} \in [0.24,0.31]$	
$p_{33} \in [10\%,20\%]$	$c_{33} \in [0.52,0.91]$	

Table B.5 Translated probabilities and costs

The costs have been transformed into the interval [0,1] by choosing the cost scale to be \$0–10 million. Now the evaluations can be carried out, using the machinery of Chapters 5–6. It is done by calculating the expected cost and expressing it as an interval. The upper bound of the interval is the maximum expected cost, and the lower bound of the interval is the minimum expected cost.

```
Probability hull      Symmetry hull
P1.1 = [0.200,0.500]  [0.243,0.500]
P1.2 = [0.100,0.200]  [0.114,0.200]
P1.3 = [0.300,0.600]  [0.343,0.600]
P2.1 = [0.300,0.500]  [0.315,0.500]
P2.2 = [0.200,0.500]  [0.223,0.500]
P2.3 = [0.150,0.300]  [0.162,0.300]
P3.1 = [0.350,0.550]  [0.350,0.532]
P3.2 = [0.300,0.550]  [0.300,0.527]
P3.3 = [0.100,0.200]  [0.100,0.191]
```

```
Value hull
V1.1 = [0.000,0.000]
V1.2 = [0.010,0.030]
V1.3 = [0.250,0.650]
V2.1 = [0.060,0.080]
V2.2 = [0.080,0.120]
V2.3 = [0.330,0.750]
V3.1 = [0.220,0.260]
V3.2 = [0.240,0.310]
V3.3 = [0.520,0.910]

Focal point
Cons.    P      V
C1.1:  0.371  0.000
C1.2:  0.157  0.020
C1.3:  0.471  0.450
C2.1:  0.408  0.070
C2.2:  0.362  0.100
C2.3:  0.231  0.540
C3.1:  0.441  0.240
C3.2:  0.414  0.275
C3.3:  0.145  0.715
```

For the actions A_1, A_2 and A_3 above expressions for the expected costs are obtained. These are denoted E_1, E_2, and E_3 respectively. For each action, both minimal and maximal expected costs have been calculated.

$\min E_1 = 0.087$
$\min E_2 = 0.110$
$\min E_3 = 0.257$

$\max E_1 = 0.395$
$\max E_2 = 0.296$
$\max E_3 = 0.407$

Table B.6 Expected costs

This means that the expected cost if action A_1 is chosen is in the interval \$870,000 to \$3,950,000. In the same way, the expected costs if actions A_2 or A_3 are chosen are in the intervals from \$1,100,000 to \$2,960,000 and \$2,570,000 to \$4,070,000 respectively. Note that these intervals are overlapping, and it seems hard to determine which action to choose based on those numbers only. Further analysis is required.

By contracting the estimates, the relationships among the three courses of action can be studied. One way is to study how the maximal and minimal expected costs behave under contraction. For a specific course of action to be better, it should have lower costs in the columns of Table B.7. Therefore, from the table it can be seen that action A_3, adding extra equipment as suggested by the security consultant, is more and more becoming the worst action the more the intervals are contracted. The overlap between A_1 and A_2 remains, however, and further analysis is necessary.

	0%	20%	40%	60%	80%
min E_1	0.087	0.109	0.132	0.158	0.186
min E_2	0.110	0.124	0.139	0.154	0.171
min E_3	0.257	0.269	0.282	0.295	0.309
max E_1	0.395	0.355	0.317	0.281	0.247
max E_2	0.296	0.273	0.250	0.229	0.208
max E_3	0.407	0.389	0.372	0.355	0.339

Table B.7 Minimal and maximal expected costs

Figures B.4–B.6 are graphic representations of the table.

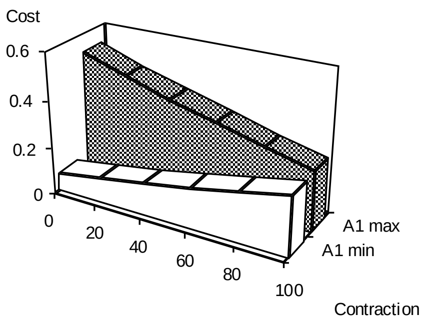

Figure B.4 Action A_1

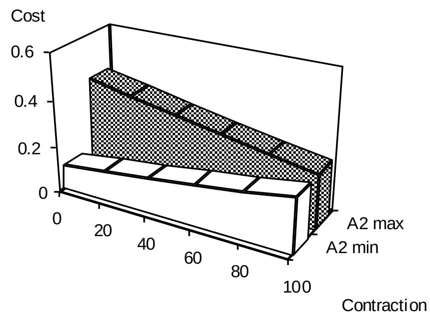

Figure B.5 Action A_2

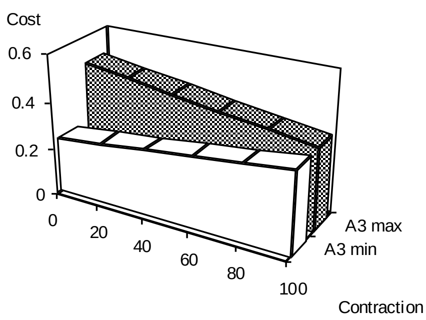

Figure B.6 Action A_3

The first evaluation was based on independent evaluation of the alternatives. The main evaluation using Δ-dominance is the next step in the DEEP evaluation. To be able to study the differences more clearly, pairwise comparisons are carried out. The results for string and weak dominance are presented in Table B.8 and illustrated in the three comparative graphs in Figures B.7–B.9. The table shows the smallest and largest difference between the courses of action. It can now more clearly be seen that action A_3 is inferior in that it is strongly NE-dominated because fairly early in the contraction process it receives positive differences, meaning it is more expensive than the others.

	0%	20%	40%	60%	80%
min (E_1-E_2)	-0.201	-0.160	-0.115	-0.069	-0.023
min (E_1-E_3)	-0.314	-0.276	-0.237	-0.197	-0.153
min (E_2-E_3)	-0.274	-0.246	-0.220	-0.196	-0.168
max (E_1-E_2)	0.284	0.231	0.178	0.127	0.076
max (E_1-E_3)	0.137	0.085	0.035	-0.014	-0.062
max (E_2-E_3)	0.038	0.002	-0.033	-0.067	-0.101

Table B.8 Pairwise comparisons between the alternatives

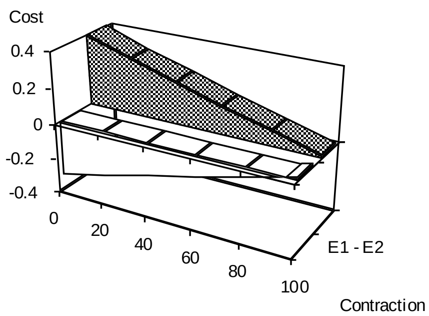

Figure B.7 Actions A_1 and A_2

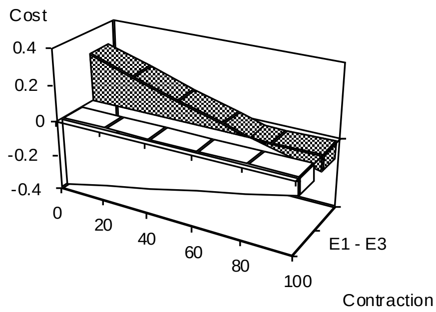

Figure B.8 Actions A_1 and A_3

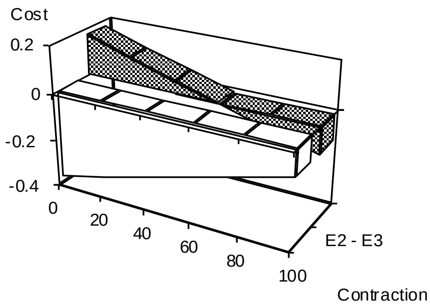

Figure B.9 Actions A_2 and A_3

To be able to discriminate between actions A_1 and A_2, further sensitivity analysis is recommended, for example by contracting subsets of intervals, not all at the same time. This will not be carried out here, since the purpose of the example is to give an impression of how DEEP can evaluate risk information. Possibly, more information is needed about the two courses of action that remain. Especially the estimates of the probabilities when burglary attempts fail are critical. If, after further analysis, it is not possible to obtain more conclusive indications, then it is an indication that the actions are indeed very similar relative to the model data. Then other activities, like contacting more equipment vendors or other insurance companies might help.

This concludes the evaluation example and the description of the DEEP method as well. A longer description can be found in [DEE96].

Every year is getting shorter
Never seem to find the time
Plans that either come to naught
Or half a page of scribbled lines

Far away across the field
The tolling of the iron bell
Calls the faithful to their knees
To hear the softly spoken magic spells

– R. Waters

References

[A53] Allais, M.: "The Foundations of a Positive Theory of Choice involving Risk and a Criticism of the Postulates and Axioms of the American School" in *Expected Utility Hypothesis and the Allais Paradox*, D. Reidel Publishing Company, 1979 (Originally in French 1953).

[B90] Bana e Costa, C.A.: *Readings in Multiple Criteria Decision Aid*, Springer-Verlag, 1990.

[BCG87] Barzilai, J., Cook, W., and Golany, B.: "Consistent Weights for Judgement Matrices of the Relative Importance of Alternatives" in *Operations Research Letters*, vol.6, pp.131–134, 1987.

[BZ70] Bellman, R., and Zadeh, L.A.: "Decision Making in a Fuzzy Environment" in *Management Science*, vol.17, pp.B-144–B-164, 1970.

[BG83] Belton, V., and Gear, A.E.: "On a Shortcoming of Saaty's Method of Analytical Hierarchies" in *OMEGA*, vol.11, pp.227–230, 1983.

[B54] Bernoulli, D.: "Exposition of a New Theory on the Measurement of Risk" in *Econometrica*, vol.22, pp.23–36, 1954 (Originally from 1748).

[B91] Bertsekas, D.: *Linear Network Optimization*, MIT Press, 1991.

[BC92] Bicchieri, C., and Chiara, M.L.D.: "Preface" in *Knowledge, Belief, and Strategic Interaction*, eds. Bicchieri and Chiara, pp.vii–xii, Cambridge University Press, 1992.

[B95] Boman, M.: "Rational Decisions and Multi-Agent Systems" in *Proc. Working Notes for the AAAI Fall Symposium on Rational Agency*, ed. Fehling, MIT Technical Report, 1995.

[BE95] Boman, M., and Ekenberg, L.: "Decision Making Agents with Relatively Unbounded Rationality" in *Proc. DIMAS'95*, pp.I/28–I/35, 1995.

[BG88] Bond, A.H., and Gasser, L.: *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann, 1988.

[BHM77] Bradley, S.P., Hax, A.C., and Magnanti, T.L.: *Applied Mathematical Programming*, Addison-Wesley, 1977.

[B84] Broder, J.F.: *Risk Analysis and the Security Survey*, Butterworth Publishers, 1984.

[B86] Brooks, R.A.: "A Robust, Layered Control System for a Mobile Robot" in *IEEE Journal of Robotics and Automation*, vol.2, pp.14–23, 1986.

[BMM88] Budnick, F.S., McLeavey, D.W., and Mojena, R.: *Principles of Operations Research in Management*, 2.ed., Irwin, 1988.

[CH92] Chen, S-J., and Hwang, C-L.: *Fuzzy Multiple Attribute Decision Making*, Lecture Notes in Economics and Mathematical Systems, vol.375, Springer-Verlag, 1992.

[C83] Chvátal, V.: *Linear Programming*, W.H. Freeman, 1983.

[CW88] Copeland, T.E., and Weston, J.F.: *Financial Theory and Corporate Policy*, 3.ed., Addison-Wesley, 1988.

[CPS92] Cottle, R.W., Pang, J-S., and Stone, R.E.: *The Linear Complementarity Problem*, Academic Press, 1992.

[C77] Courtney, R.H.: "Security Risk Assessment in Electronic Data Processing" in *AFIPS NCC*, vol.46, 1977.

[D93] Danielson, M.: Implementation av ett system för beslutsanalys, WP-196, Dept. of Computer and Systems Sciences, Royal Institute of Technology, 1993.

[D95] Danielson, M.: Computing Best Choices using Imprecise Information, Licentiate Thesis, Dept. of Computer and Systems Sciences, Royal Institute of Technology, 1995.

[D96] Danielson, M.: DDT – the DELTA Decision Tool, Research Report 96-026, Dept. of Computer and Systems Sciences, Royal Institute of Technology, 1996. Proceedings of the IIASA workshop on Advances in Methodology and Software for Decision Support Systems and Software, Laxenburg, Austria, September 1996.

[D97a] Danielson, M.: DDT dwish Specification, version 2.4, Dept. of Computer and Systems Sciences, Royal Institute of Technology, 1997.

[D97b] Danielson, M.: Different Research Directions and Approaches to Interval Decision Analysis within the DECIDE Research Group, Dept. of Computer and Systems Sciences, Royal Institute of Technology, 1997.

[DE97a] Danielson, M., and Ekenberg, L.: "Evaluating Decision Trees under Different Criteria", Proceedings of 13th Int. Conf. on Multiple Criteria Decision Aids, Cape Town, 1997.

[DE97b] Danielson, M., and Ekenberg, L.: Riskbedömning i vaga domäner, NUTEK Project Report P5873-1, 1997.

[DE97c] Danielson, M., and Ekenberg, L.: "A Framework for Analysing Decisions under Risk" to appear in *European Journal of Operational Research*, 1997.

[DEE96] Danielson, M., Ekenberg, L., and Elgemyr, A.: "Riskanalys med DEEP-metoden" in *Scandinavian Insurance Quarterly*, vol.77, no.4, pp.311–324, 1996.

[D67] Dempster, A.P.: "Upper and Lower Probabilities Induced by a Multivalued Mapping" in *Annals of Mathematical Statistics*, vol.XXXVIII, pp.325–339, 1967.

[D90] Dixon, G.: *Riskanalys*, SBF – Svenska Brandförsvarsföreningen, 1990.

[D92] Doyle, J.: "Rationality and its Roles in Reasoning" in *Computational Intelligence*, vol.8, no.2, pp.376–409, 1992.

[E77] Edwards, W.: "How to Use Multi-Attribute Utility Measurement for Social Decisionmaking" in *IEEE Transactions on Systems, Man, and Cybernetics*, vol.7, no.5, pp.326–340, 1977.

[E94] Ekenberg, L.: Decision Support in Numerically Imprecise Domains, Ph.D. Thesis, Dept. of Computer and Systems Sciences, Stockholm University, 1994.

[EBD95] Ekenberg, L., Boman, M., and Danielson, M.: "A Tool for Co-ordinating Autonomous Agents with Conflicting Goals" in *Proceedings of the 1st International Conference on Multi-Agent Systems ICMAS '95*, pp.89–93, AAAI/MIT Press, 1995.

[ED94] Ekenberg, L., and Danielson, M.: "A Support System for Real-Life Decisions in Numerically Imprecise Domains" in *Operations Research Proceedings 1994*, Berlin, Germany, eds. Derigs, Bachem, and Drexl, pp.500–505, Springer-Verlag, 1994.

[ED95] Ekenberg, L., and Danielson, M.: "Handling Imprecise Information in Risk Management" in *Information Security – the Next Decade*, eds. Eloff and von Solms, pp.357–368, Chapman & Hall, 1995.

[EDB96a] Ekenberg, L., Danielson, M., and Boman, M.: "A Tool for Handling Uncertain Information in Multi-Agent Systems" in *Distributed Software Agents and Applications – Proceedings of the MAAMAW '94*, eds. Perram and Müller, pp.54–62, Springer-Verlag, 1996.

[EDB96b] Ekenberg, L., Danielson, M., and Boman, M.: "From Local Assessments to Global Rationality" in *International Journal of Cooperative Information Systems*, vol.5, nos.2&3, pp.315–331, 1996.

[EDB97] Ekenberg, L., Danielson, M., and Boman, M.: "Imposing Security Constraints on Agent-Based Decision Support" to appear in *Decision Support Systems International Journal*, 1997.

[EM92] Elgemyr, A., and Mattsson, L.: *Stora säkerhetsboken*, Publica, 1992.

[ESF91] European Security Forum: *A Risk Analysis Method which is Easy to Understand and Simple to Apply*, Draft Method, 1991.

[F70] Fishburn, P.C.: *Utility Theory for Decision Making*, John Wiley and Sons, 1970.

[F81] Fishburn, P.C.: "Subjective Expected Utility: A Review of Normative Theories" in *Theory and Decision*, vol.13, pp.139–199, 1981.

[F83] Fishburn, P.C.: "Transitive Measurable Utility" in *Journal of Economic Theory*, vol.31, pp.293–317, 1983.

[F80] Freeling, A.N.S.: "Fuzzy Sets and Decision Analysis" in *IEEE Transactions on Systems, Man, and Cybernetics*, vol.10, no.7, pp.341–354, 1980.

[F88] French, S.: *Decision Theory: An Introduction to the Mathematics of Rationality*, Ellis Horwood, 1988.

[FH84] Fuller, R.J., and Hsia, C-C.: "A Simplified Common Stock Valuation Model" in Financial Analysts Journal, Sep.-Oct. 84, pp.49–56, 1984.

[G86] Galliers, J.R.: "A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict" in *IEEE Journal of Robotics and Automation*, vol.2, pp.14–23, 1986.

[GN87] Genesereth, M.R., and Nilsson, N.J.: *Logical Foundations of Artificial Intelligence*, Morgan Kaufmann, 1987.

[GD93] Gmytrasiewicz, P.J., and Durfee, E.H.: "Elements of a Utilitarian Theory of Knowledge and Action" in *Proceedings of 13th IJCAI*, pp.396–402, 1993.

[GT89] Goldfarb, D., and Todd, M.J.: "Linear Programming" in *Optimization*, Handbooks in Operations Research and Management Science, vol.1, eds. Nemhauser, Rinnooy Kan, and Todd, Elsevier, 1989.

[G62] Good, I.J.: "Subjective Probability as the Measure of a Non-measurable Set" in *Logic, Methodology, and the Philosophy of Science*, eds. Suppes, Nagel, and Tarski, pp.319–329, Stanford University Press, 1962.

[G92] Gonzaga, C.C.: "Path-Following Methods for Linear Programming" in *SIAM Review*, vol.14, no.2, pp.167–224, Society for Industrial and Applied Mathematics, 1992.

[G92b] Green, B.: "Vad kan bankerna lära sig av en entrepenör som utvecklas till organisationsforskare" in *Riskbedömning – kunskap om risker*, NUTEK, Stockholm, pp.121–126, 1992.

[GS82] Gärdenfors, P., and Sahlin, N-E.: "Unreliable Probabilities, Risk Taking, and Decision Making" in *Synthese*, vol.53, pp.361–386, 1982.

[H60] Halmos, P.R.: *Naive Set Theory*, D. van Nostrand Co., 1960.

[H88] Hamilton, G.: *This is Risk Management*, Chartwell-Bratt, 1988.

[HM53] Hernstein, I.N., and Milnor, J.: "An Axiomatic Approach to Measurable Utility" in *Econometrica*, vol.21, pp.291–297, 1953.

[HM84] Howard, R.A., and Matheson, J.E.: "Influence Diagrams" in *Principles and Applications of Decision Analysis*, eds. Howard and Matheson, vol.II, Strategic Decisions Group, Menlo Park, CA, USA, 1984.

[H89] Hull, J.: *Options, Futures, and Other Derivative Securities*, Prentice-Hall, 1989.

[H51] Hurwicz, L.: *Optimality Criteria for Decision Making under Ignorance*, Cowles Commission Discussion Paper no.370, 1951.

[J88] Jarrow, R.A.: *Finance Theory*, Prentice-Hall, 1988.

[J83] Jeffrey, R.: *The Logic of decision*, University of Chicago Press, 1983.

[K92] Keeney, R.L.: *Value-Focused Thinking: A Path to Creative Decision Making*, Harvard University Press, 1992.

[KR76] Keeney, R.L., and Raiffa, H.: *Decisions with Multiple Objectives: Preferences and Value Trade-offs*, John Wiley and Sons, 1976.

[K96] Kemikontoret: Riskhantering 1, Administrativ SHM-revision, 4.ed, 1996.

[K90] Kreps, D.M.: *A Course in Microeconomic Theory*, Harvester Wheatsheaf, 1990.

[K87] Krovak, J.: "Ranking Alternatives – Comparison of Different Methods Based on Binary Comparison Matrices" in *European Journal of Operational Research*, vol.32, pp.86–95, 1987.

[LH94] Lai, Y-J., and Hwang, C-L.: *Fuzzy Multiple Objective Decision Making*, Lecture Notes in Economics and Mathematical Systems, vol.404, Springer-Verlag, 1994.

[L25] Laplace, P.: *Essai Philosophique sur les Probabilites*, 5.ed., Paris, 1825. (Translation published by Dover 1952.)

[LF90] Lee, C.F., and Finnerty, J.E.: *Corporate Finance: Theory, Method, and Applications*, Harcourt Brace Jovanovich, 1990.

[L59] Lehmann, E.L.: *Testing Statistical Hypothesis*, John Wiley and Sons, 1959.

[L74] Levi, I.: "On Indeterminate Probabilities" in *The Journal of Philosophy*, vol.71, pp.391–418, 1974.

[L92] Levi, I.: "Feasibility" in *Knowledge, Belief, and Strategic Interaction*, eds. Bicchieri and Chiara, Cambridge University Press, pp.1–20, 1992.

[LS82] Loomes, G., and Sugden, R.: "Regret Theory: An Alternative Theory of Rational Choice under Uncertainty" in *The Economic Journal*, vol.92, pp.805–924, 1982.

[L93] Lootsma, F.A.: "Scale Sensitivity in the Multiplicative AHP and SMART" in *Journal of Multi-Criteria Decision Analysis*, vol.2, pp.87–110, 1993.

[L89] Luenberger, D.G.: *Linear and Nonlinear Programming*, 2.ed., Addison-Wesley, 1989.

[M81] Malmnäs, P-E.: *From Qualitative to Quantitative Probability*, Ph.D. Thesis, Almqvist & Wiksell International, 1981.

[M90] Malmnäs, P-E.: Real-Life Decisions, Expected Utility, and Effective Computability, Research Report HSFR no.677/87, 1990.

[M94a] Malmnäs, P-E.: "Towards a Mechanization of Real Life Decisions" in *Logic and Philosophy of Science in Uppsala*, eds. Prawitz and Westerståhl, Kluwer Academic Publishers, 1994.

[M94b] Malmnäs, P-E.: "Axiomatic Justifications of the Utility Principle" in *Synthese*, vol.99, no.2, pp.233–249, 1994.

[M96] Malmnäs, P-E.: Evaluations, Preferences, and Choice Rules, Dept. of Philosophy, Stockholm University, 1996.

[MM73] Mason, R.O., and Mitroff, I.I.: "A Program for Research on Management Information Systems" in *Management Science*, vol.19, no.5, 1973.

[M92] McClennen, E.F.: "Rational Choice in the Context of Ideal Games" in *Knowledge, Belief, and Strategic Interaction*, eds. Bicchieri and Chiara, pp.47–60, Cambridge University Press, 1992.

[M34] Menger, K.: "Das Unsicherheitsmoment in der Wertlehre" in *Zeitschrift für Nationalökonomie*, vol.5, pp.49–59, 1934.

[M54] Milnor, J.: "Games against Nature" in *Decision Processes*, eds. Thrall, Coombs, and Davis, pp.49–59, John Wiley and Sons, 1954.

[N90] Neapolitan, R.E.: *Probabilistic Reasoning in Expert Systems: Theory and Algorithms*, John Wiley and Sons, 1990.

[NM47] von Neumann, J., and Morgenstern, O.: *Theory of Games and Economic Behavior*, 2.ed., Princeton University Press, 1947.

[N81] Newell, A.: "The Knowledge Level" in *AI Magazine,* Summer Issue, pp.1–20, 1981.

[N86] Nilsson, N.J.: "Probabilistic Logic" in *Artificial Intelligence*, vol.28, pp.71–87, 1986.

[OM90] Oddie, G., and Milne, P.: "Act and Value" in *Theoria*, vol.LVII, pp.42–76, 1990.

[O96] Olson, D.L.: *Decision Aids for Selection Problems*, Springer-Verlag, 1996.

[P91] Pearl, J.: *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, revised 2nd printing, Morgan Kaufmann, 1991.

[Q82] Quiggin, J.: "A Theory of Anticipated Utility" in *Journal of Economic Behavior and Organisation*, vol.3, pp.323–343, 1982.

[R92] Rabinowicz, W.: "Tortuous Labyrinth: Noncooperative Normal-Form Games between Hyperrational Players" in *Knowledge, Belief, and Strategic Interaction*, eds. Bicchieri and Chiara, Cambridge University Press, pp.107–126, 1992.

[R68] Raiffa, H.: *Decision Analysis: Introductory Lectures on Choices under Uncertainty*, Random House, 1968.

[R78] Ramsey, F.P.: "Truth and Probability" in *Foundations: Essays in Philosophy, Logics, Mathematics and Economics*, ed. Mellor, pp.58–100, Routledge & Kegan Paul, 1978.

[R93] Rosenschein, J.S.: "Consenting Agents: Negotiation Mechanisms for Multi-Agent Systems" in *Proceedings of 13th IJCAI*, pp.792–799, 1993.

[RS95] Russell, S.J., and Subramanian, D.: "Provably Bounded-Optimal Agents" in *Journal of Artificial Intelligence Research*, vol.2**,** pp.575–609, 1995.

[S80] Saaty, T.L.: *The Analytical Hierarchy Process*, McGraw-Hill, 1980.

[SAF86] SAF: *Riskanalys*, Näringslivets Beredskapsbyrå, 1986.

[SW84] Sage, A.P., and White, C.C.: "ARIADNE: A Knowledge-Based Interactive System for Planning and Decision Support" in *IEEE Transactions on Systems, Man, and Cybernetics*, vol.14, no.1, 1984.

[SH95] Salo, A.A., and Hämäläinen, R.P.: "Preference Programming through Approximate Ratio Comparisons" in *European Journal of Operational Research*, vol.82, pp.458–475, 1995.

[S51] Savage, L.: "The Theory of Statistical Decision" in *Journal of the American Statistical Association*, vol.46, pp.55–67, 1951.

[S72] Savage, L.: *The Foundations of Statistics*, 2.ed, Dover, 1972.

[S82] Schoemaker, P.: "The Expected Utility Model: Its Variants, Purposes, Evidence, and Limitations" in *Journal of Economic Literature*, vol.XX, pp.529–563, 1982.

[S86] Shachter, R.D.: "Evaluating Inference Diagrams" in *Operations Research*, vol.34, no.6, 1986.

[S76] Shafer, G.: *A Mathematical Theory of Evidence,* Princeton University Press, 1976.

[SP89] Shafer, G., and Pearl, J., eds.: *Readings in Uncertain Reasoning*, Morgan Kaufmann, 1989.

[S76b] Simon, H.A.: *Administrative Behaviour*, 3.ed. The Free Press, New York, 1976.

[S61] Smith, C.A.B.: "Consistency in Statistical Inference and Decision" in *Journal of the Royal Statistic Society*, Series B, vol.XXIII, pp.1–25, 1961.

[S89–91] Statskontoret: *Vägledning i ADB-säkerhet 1–8*, 1989–91.

[V92] Vincke, P.: *Multicriteria Decision Aid*, John Wiley and Sons, 1992.

[W50] Wald, A.: *Statistical Decision Functions*, John Wiley and Sons, 1950.

[W97] Walter, J.: A Decision Tool for Uncertain Decision Making, Master's Thesis, Dept. of Numerical Analysis and Computing Science, Stockholm University, 1997.

[WF82] Watson, S.R., and Freeling, A.N.S.: "Assessing Attribute Weights" in *OMEGA*, vol.10, pp.582–583, 1982.

[WP90] Weichselberger, K., and Pöhlman, S.: *A Methodology for Uncertainty in Knowledge-Based Systems*, Springer-Verlag, 1990.

[W91b] Wermdalen, H.: *Securitas – Säkerhetsboken 1992*, Studentlitteratur, 1991.

[W84] Wrede, R.: "The SBA Method: A Method for Testing Vulnerability" in *Proc. IFIP/SEC '84*, pp.313–320, 1984.

[W91] Wright, M.H.: "Interior Methods for Constrained Optimization" in *Acta Numerica*, pp.341–407, 1991.

[Y87] Yaari, M.: "The Dual Theory of Choice under Risk" in *Econometrica*, vol.55, pp.95–115, 1987.

[Z96] Zilberstein, S.: "Using Anytime Algorithms in Intelligent Systems" in *AI Magazine*, Fall 1996, pp.73–83, 1996.

[ZZG84] Zimmermann, H.J., Zadeh, L.A., and Gaines, B.R.: *Fuzzy Sets and Decision Analysis,* TIMS Studies in the Management Sciences, vol.20, North-Holland, 1984.