Uncertainty in Planning: Adapting the Framework of Game Theory

David J. Marsay
Defence Evaluation and Research Agency
St. Andrews Road
Malvern
WORCS, UK WR14 3PS
djmarsay@dera.gov.uk

Abstract

AI and AI-aided planning are currently at their best where one has a large, but not too large, number of possible actions that interact with a well defined 'problem space' in well defined ways. Many applications are not quite like this. The last few years have seen a surge of interest in the 'Bayesian' approach to uncertainty, but this has not been as fruitful as had been hoped. This paper adapts the framework of game theory to represent some common types of uncertainty, even where there is no gaming element in the problem. This framework gives different results from a conventional utility-maximization approach. Familiar route-planning examples are used to show that the new game-theoretic framework can be applied to a significant range of problems where utility maximization has significant defects. The implications of this are briefly discussed.

Introduction

At its simplest, the AI planning problem is to find a sequence of actions that will take the planning world from a defined initial condition to a given goal state. It is essentially a puzzle, with the challenge generally provided by the great number of possible alternative plans. A typical example is route planning. This is often idealized into a simple puzzle, of finding a route that is 'reasonable', in some sense. However, real-world planning often involves a significant element of uncertainty, such as uncertain weather or traffic. There is not yet a consensus within the AI planning community on how these uncertainties should be handled. The probabilistic (Bayesian) approach has been widely studied, but there seems to be no single approach that is widely applicable. This was discussed in the summer 1999 AI Magazine special issue on Bayesian Techniques, e.g. (Haddawy, 1999).

This article gives some general insights into uncertainty in planning, and suggests that the framework and methods of game theory can be used, even where the assumptions of game theory - that there is a game - do not hold. It also argues that this novel approach to planning is more widely applicable than those based on conventional 'Bayesian' utility maximization.

Game theory

The conventional game theory formulation is where each player has a number of possible actions, and each set of choices of actions by the players has a consequence, whose 'utility' is typically different for each player. The conventional strategy is for each player to choose that action for which the worst outcome over all the other player's assignments is best, or least bad (von Neumann & Morgenstern 1944). This framework can be used to capture uncertainties in the situation, such as the weather, that are not due to an 'opponent', and the solution that game theory gives is often satisfactory. However, game theory does not currently play any wider explicit role in planning. (Russell & Norvig 1995)

Route Planning example

Consider a route planning problem, where we need to plan a route for use tomorrow morning. We classify the potential weather as 'fair', 'wet' or 'foggy', say. For example, fair weather is any weather that does not impede us. We want to reach our destination at an agreed time, and to be able to start our journey as late as possible - so that we do not have to get up too early.

The times that we allow for a given journey depend on the route and the weather as shown below:

	Fair	Wet	Fog	Worst	Average
Motorway	100	120	200	200	140
Major roads	120	150	180	180	150
Minor roads	150	190	170	190	170

Times in minutes

These notional figures are based on a situation in which we can find a shorter route by using minor roads. The better roads have the benefit of higher speeds, but this is lost in bad weather. In this example the minor roads can be subject to partial flooding. The example is contrived to illustrate some of the main issues, but is intended to be 'representative'. The average is based on a situation where each weather condition has been judged to be equally likely.

Using the motorway minimizes the expected journey time. This is the Bayesian route. But to be sure of arriving on time those following the Bayesian route must allow 200 minutes, in case of fog. If we plan a route by major roads we can start 20 minutes later, since the worst case delay is less. This is the plan that we would choose if we thought that the weather was playing a game against us.

Suppose now that we are sure that when we get up we will know the weather. If the weather turns out to be fine we can always go by motorway, thus not wasting time. If we also plan a minor road route, to be used in case of fog, we can save a further 10 minutes, since the worst case where we choose our route knowing the actual weather is 170 minutes, a saving of 30 minutes, 15%, over the Bayesian plan. This corresponds to determining the worst-case utility from playing a given game, not selecting an actual play.

This simple example demonstrates that the framework of game theory can be used to select a plan that is clearly better than the Bayesian plan. Moreover, there is no need to estimate the probabilities of the particular conditions, and the plan will be much more stable from day to day, only changing when we judge that the possibility of fog, say, can be discounted.

It might seem from this that we have a counter-example to Bayes’ decision theory (Bayes 1763). Actually, we have simply shown that his preconditions do not apply. Planning is not like gambling. In this paper the term ’Bayesian’ is used to describe the application of methods based on Bayes’ theory without regard to the preconditions.

General approach

Notation

A generalization of the above approach is now developed using a pseudo-code approach based on (Russell & Norvig 1995). As a convenience, SUM(,) is first defined by:

function SUM(*FUNCTION*, *space*)
// This is a comment
sum ← ZERO
for each element **in** *space*
sum ← sum + *FUNCTION*(element))
return sum

Thus SUM(,) has variables *FUNCTION* and *space* and returns the sum of *FUNCTION*( ) over *space*. MIN(,) is defined similarly, but taking the minimum value. Functions are shown upper-case, sets are underlined. Sets of functions will be upper-case underlined. If we have ’subset ⊆ set’, say, this is sometimes represented by the function ’SUBSET: set → {0, 1}’. Other functions are denoted similarly, where it is not appropriate to define their bodies.

We assume that we have a ’given’ set of situations, situation, a set of possible plans, plan, and some ’figure of merit’ for plans in specific situations: ’FOM: plan × situation → value’. This is the same formalization as in game theory, and generalizes that of Bayesian decision theory. This figure of merit could reflect, for example, ’time to be allowed’, ’fuel to be consumed’ and ’risk of accident’. Here we are not concerned with how to derive a figure of merit, but only how to use it in planning. In a deterministic situation we simply maximize the figure of merit. But how should we compare plans when the situation is uncertain?

Evaluation

According to Bayesians, we should estimate the probabilities ’P: situation → [0,1]’. Then BAYESIAN-UTILITY(,) sums the ’value’ of the plan in a situation times the probability of the situation, over all situations, thus:

function BAYESIAN-UTILITY(*plan*, *P*)
// *P* is a Bayesian probability
return SUM((FOM(*plan*,)**P*( )), situation)

The route planning example shows that this is not always optimal. It disregards Bayes’ preconditions, so why not do the same for game theory? Consider a set of possible situations in a particular case, possible, as if they were chosen by an opponent. Then we have:

function POSSIBILISTIC-UTILITY(*plan*, *possible*)
return MIN(FOM(*plan*,), *possible*)

This is simpler than the Bayesian version, and the example has shown that it can give correct results where the Bayesian one fails significantly. But there are familiar examples where the converse is true, such as gambling. To deal with those suppose that we have a generalized probability, GPROBABILITY( ), (e.g, PROBABILITY) and a corresponding evaluator GUTILITY(,) (e.g., BAYESIAN-UTILITY). Consider a new entity, ’possible probability assignments’, ’ppa ⊆ GPROBABILITY’. Then this is a genuine generalization, since a singleton ppa corresponds to a unique (generalized) probability and a ppa in which each situation maps into {0,1} corresponds to a unique set of possibilities. It can thus deal with a ’real’ route planning problem where even on a particular journey there is still an element of subjective probability in which roads will be busy. So this extension is more like the ’product of the parts’ than the simple sum.

The corresponding evaluator is

function EU(*plan*, *PPA*)
// *PPA* is a possible probability assignment
return MIN(GUTILITY(*plan*,), *PPA*)

It may be we have no means to evaluate GUTILITY(,) directly. In these cases we can generate a representative set of deterministic test cases, in effect a ’simulation’. For example, if each uncertain element is a simple binary ’yes’ or ’no’, as in road open or closed, one can use (informally):

```
function SAMPLE_CASE(P)
// P is a probability function
  for each element in situation
    part(element) ← 'RAND() < P (element)'
  deterministic_situation ← PROD(part, situation)
return deterministic_situation
```

In the familiar Monte-Carlo method one averages the deterministic utilities to estimate the Bayesian utility. For different utilities such as EU(,) one simply uses the appropriate function, in this case MIN(,).

Plan Refinement

There will often be many potential plans that could be evaluated. One often needs a heuristic to rapidly identify a relatively small set of candidate plans. An important feature of EU(,) is that we can find the set of worst-case situations with the following general heuristic :

```
function WORST_CASE(plan, PPA)
  best ← EU(plan, PPA)
  worst_cases ← NULL
  for each situation in situation
    if EU(plan, {situation}) == best
      then worst_cases ← worst_cases ∪ {situation}
return worst_cases
```

Above, the term '{situation}' denotes the singleton set containing 'situation'. This evaluator can be used to improve a plan using a given a set of planners, PLANNER (for example, containing the Bayesian planner) as follows:

```
function REFINE(plan, PPA)
  // NOT optimised in any sense
  best_value ← EU(plan, PPA)
  for each PLANNER in PLANNER do
    for each problem in WORST_CASE(plan, PPA) do
      candidate_plan ← PLANNER(problem)
      value ← EU(candidate_plan, PPA)
      if value > best_value then do
        plan ← candidate_plan
        best_value ← value
      end
    end
  end
return plan
```

This is then iterated in the usual way:

```
function IMPROVE (plan, PPA)
  loop do
    old_plan ← plan
    plan ←  REFINE(plan, PPA)
  until (plan == old_plan)
// or one could terminate earlier
return plan
```

This is the familiar worst-case heuristic. In the route planning example, 'fog' is the worst case, so one considers the 'minor road' plan. This route is worse for 'wet' weather, so one looks for a plan that is good in 'fog' and 'wet'. This is 'major roads', as obtained previously.

Note that the method has reduced the number of situations and plans that have to be considered. We did not even need to evaluate the Bayesian route. This is in contrast to the Bayesian approach, where at least the Bayesian route has to be evaluated for every condition. In general we will look for a domain-specific HEURISTIC(*plan*, *PPA*) (possibly independent of *plan*) to reduce the number of situations that have to be considered. This will be essential where PPA is very large or infinite. In practice we may have to use an approximate HEURISTIC, so that for a plan, pl, under consideration,

EU(*pl*, *PPA*) ≅ EU(*plan*, HEURISTIC(*pl*, *PPA*). When using an approximation it will be a useful safeguard to validate the iterated plan against the whole PPA. Hence the use of an approximation need only serve to facilitate the search without invalidating the result.

Should we consider the iterated plan to be a good one? Where the Bayesian plan is considered to be reasonable, one approach is to consider a plan to be acceptable if it is not much worse than the Bayesian plan in terms of the Bayesian criterion. For problems like route planning, this will be the case provided that the uncertain conditions are not too restrictive and the route network is not too sparse But other planning problems may have more critical dependencies.

Planning in a Game-Theoretic Framework

In game theory, there are normally only a relatively few possible actions, and they are normally 'given'. In planning there are potentially a huge number of possible plans (e.g., routes) and (as in many real-life games) the problem is to generate candidate plans for evaluation. This creates complications, but they are not insurmountable.

Assignment Spaces

We need a way of describing a space of possible plans and their relationships to possible outcomes that can be used before the plans, outcomes and uncertainties have been identified. The description of such a space could clearly be quite complex. To be able to generate and evaluate plans we need a pragmatic way forward. Most current notions of generalized probability, including comparative probability (von Neumann & Morgenstern 1944) and imprecise probability (Walley 1991) can be represented by constraints on Bayesian probability assignments. Thus the exposition can be simplified by supposing that our 'possible probability assignments' take this form. This allows us to represent scenarios, such as in the route planning case, where any given situation can be represented by Bayesian probabilities, but we do not know which situation will actually occur. We may be able to say that we have a 95% chance of arriving on time if we use a given route in the wet, but we don't yet know what the weather will be like tomorrow morning.

If the constraints on values are linear (as in comparative or imprecise probabilities) then the resultant space is non-empty (provided the constraints are consistent) and forms the inside of a convex polyhedron. Such a space can be relatively compactly described by giving the vertices. The whole space is simply the convex hull of the vertices. Walley (Walley 1991, 3.6) has generalized this, to show that 'previsions' that avoid 'sure loss' are the lower envelope of the set of 'extreme points' of 'dominating linear previsions'. This can be used to identify 'good' plans.

This 'assignment space' approach can be used with the probabilities being probabilities of success of actions. We then maximize the worst case probability of success. Alternatively, the assignments could be of the usage of resources that relate to overall utility.

In a straightforward application, a vertex will typically correspond to a particular combination of factors such as 'weather' and 'time of day'. This could be used directly if the values can simply be interpolated between extreme values. If not, it is necessary to insert intermediate conditions until we have a reasonable approximation.

The dependent variables could relate to every possible situation individually., but one would normally seek some compactness. For example in a route planning example one could classify road elements and associate common measures (such as length) with them. The condition-dependent parameters could then be used with the road-element values to give overall resource usage. For example, for a condition 'dry weather, night' one could have a list of notional speeds against road type. Thus the examples presented above may all be dealt with by natural assignment spaces. Plans are evaluated and refined as describe above.

An assignment space could also be used to record the space of 'base' probabilities to be used in a Bayesian net (D'Ambroise 1999). However such a net is not useful, since the preconditions for practical updating do not hold[1]. An alternative approach to plan generation is therefore required.

Plan generation

Plan generation typically depends on domain-dependent heuristics. However, there are some general observations that can be made, that lead to improved plans

A simple way to generate more plans from an existing set of candidates is to split them into fragments and recombine. This is often a satisfactory heuristic, provided that the initial set of plans is sufficiently diverse. In route planning this approach can often generate a good choice of routes to be evaluated, but it is not 'directed', and hence not efficient unless evaluation is cheap.

Now, consider three routes: one good in the morning rush, one good in the afternoon rush, and one a good compromise. Looking only at vertices or even at 'point' probability assignments we may never find the compromise. A common approach is to plan against conditions that are averaged over the day. But, since the rush periods are relatively short, this will tend to find the overall quickest route under good conditions, which may not be the best compromise. 'Bayesian' planning is clearly not appropriate under these conditions.

An obvious heuristic, at least for route planning, is 'TEST_CASE( )', which breaks a <u>situation</u> into elements (such as the states of the individual roads), for each element finds the worst case over the possible probability assignments, *PPA*, and then reconstitutes the generalized probability for the overall <u>situation</u>.

function TEST_CASE(*PPA*)
  for each element **in** <u>situation</u>
    TMP(element)) ← WORST(*PPA*|element)
  GPROB (<u>situation</u>) ← PROD(TMP (), <u>situation</u>)
return GPROB

Here, 'PPA|element' denotes the set of probability assignments, PPA, restricted to 'element'. PROD(,) denotes the product over the set, like SUM(,). WORST(element, ,) will typically be MIN(,) or MAX(,), depending on the element.

In terms of route planning, this will give a good compromise route if there is one. This is typical of many current AI planning problems, where there is generally a good plan, with AI technology being used to speed up the search. However, in some cases it may be that there is no such compromise route. One may then need to consider more complex plans than 'use route X', such as observing the traffic at a motorway junction before committing to the rest of the route. Furthermore, close attention should be given to the 'given' figure of merit, to make sure that it is valid. These aspects are beyond the scope of this paper, although the evaluation and refinement approaches still apply.

Currently, Bayesian planners are often not practical. Hence the TEST_CASE(,) cannot be used. SAMPLE_CASE() (above) can be used, though. The general approach is practical in a small problem such as the route example, where there are only three possible weather conditions, but is obviously limited. What if one is travelling a long way and wants to consider the result of possible accidents?

Performance

The main aim in this paper is in establishing the correct criteria for evaluating plans, and showing how to select good plans. The details of the actual planner will be very domain-dependent, and would normally incorporate domain-dependent heuristics, so all one can do in a general way is to show that existing planners

[1] Typical rules for updating of Bayesian nets depend on both Bayes' rule and Bayes' observation that the probability of one or other of two disjoint possibilities is the sum of their probabilities (in effect, 'probabilities sum to 1'). Their generalizations (Walley 1991) are too imprecise to give efficient updating, even in our simple example.

can be re-used within the 'new' concepts.

In some cases, though, the approach outlined above will actually give an optimal plan. If the utilities are monotonic, so that the extremes are at the possibilistic vertices, then this method will give a definite best plan. Even where one has more complex utilities, there are tests that can be used to show that the result is 'reasonable' in some absolute sense, and not just the best of those considered. If one has a Bayesian plan, and if the process described gives a plan that is not significantly worse than the Bayesian plan according to the Bayesian criterion, then the plan is clearly absolutely reasonable. The reason for this is the Bayesian plays a role as a 'benchmark'. If one simply has a collection of plans with no benchmark, there is no way to establish absolute performance.

Requirements

It is not envisaged that the techniques envisaged here will be incorporated into specialist planners for every domain. Instead, one will have specialist tools that act as Oracles, used within a larger planning process, as described above. However this implies certain capabilities on the part of the specialist tools:

- A tool (planner) to accept notional worst-case conditions (even if not physically feasible - e.g. tracks may simultaneously slow down vehicles that are susceptible to dust as well as those susceptible to mud.)
- A tool (e.g., simulator) to evaluate any given route against a specified condition. (This might be deterministic, Monte-Carlo or probabilistic.)

MicroSoft's familiar AutoRoute route illustrates the issues. One can define speed on each class of road, and thus can plan against either a particular condition (weather and traffic) or a notional worst case condition. Ideally, we would want more types, for example to indicate routes that are susceptible to flooding or fog, but the principle is there.

AutoRoute can also be used (with 'snap routing'). to evaluate a given route. It also provides the ability to block selected areas. This could be used as a means of checking for robustness, but it would not be practical to do this manually. AutoRoute also has the ability to generate new routes using fragments of existing routes, combining them where they cross. It can thus potentially generate a large number of routes to be evaluated. The main limitation of AutoRoute is that it does not currently accept timely updates on relevant data from, say, TrafficMaster PLC's TrafficMaster system.

In some cases one might have two or more tools that do a similar job, but have different strengths or specializations. Combining such tools may be thought of as reducing uncertainty. The requirements called for above are clearly necessary to enable tools with different specializations to be combined. For example, if one tool knows about the effects of flooding on roads and the other knows about traffic patterns (e.g., knows when school terms are) then we need a similar process to use the tools to generate a reasonable route.

Related Work and Discussion

This new approach essentially complements mainstream work on deterministic planning and Bayesian planning - e.g. (Blythe 1999, Haddawy, 1999). It brings to planning more general insights on making decisions concerning an uncertain future (Smith 1776, Bernstein 1996). Previous work has shown the limitations of 'Bayesian' probability in other domains (Keynes 1921, Smith 1961, Soros 1996, Skyrms 2000, Walley 1991), including strategic defence planning (Enthoven & Smith 1971), but not for the type of planning that is typically the subject of AI planning.

(Cheeseman 1985) provides a benchmark view in defence of the use of conventional probability in AI in general, and the adequacy of 'Bayesian' probability in AI is now generally taken almost for granted (e.g., Russell & Norvig 1995, Blythe 1999). This paper is the first to give an accessible planning example of the inadequacy of 'Bayesian probability'. It is also the first to adapt the methods of generalized probability (Walley 1991) to planning, and to put these methods in the context of game theory.

In the wider context, this paper is unique in that elsewhere the inadequacy of the Bayesian approach is accounted for either solely by the variability in the situation or by the effect of other human decision-makers (Bernstein 1996). What has been shown here is that the Bayesian approach is not appropriate when we wish to plan actions that facilitate further plans.

The Bayesian approach is appropriate where the assumptions of Bayes' decision-theory hold[2]. Where one has Bayesian probabilities, one has to make a definite plan now for future enactment, and one wishes to maximize some long-run utility, averaged over many plans. In the route-planning example there are two problems. Firstly, the appropriate utility is not the 'Bayesian' norm. Secondly, we have the opportunity to change the route when we see what the actual weather is. But in these examples the underlying 'objective' probabilities may still be 'Bayesian'.

The game-theoretic framework is powerful, but it needs to be shown that it does not exclude other desirable ways of extending conventional planning. Here, following (Bernstein 1996), three extensions are considered: robustness analysis and coping with domains that exhibit 'non-linearity' or 'reflexivity'.

Robustness analysis estimates how robust a plan might be to actions becoming unavailable, for example to roads being blocked. This could be deterministic or Bayesian. To include this within the framework the utility would need to be modified. For example, one might take the cost of travelling from A to B as taking into account possible detours. (This could be done using SAMPLE_CASE( ) to test self-repairing plans).

Non-linearity is where costs do not simply accumulate. For example, if I am planning to skirt a

[2] By Bayesian I mean like Bayes, but ignoring the preconditions. There is no conflict between Adam Smith and Bayes.

town while the morning rush hour is building up, the effects of any early delays can become greatly magnified. When going through a large city around rush hour, the likely distribution of delays can have two distinct 'modes': before and after the rush hour peak. This non-linearity complicates the derivation of utility, but that does not invalidate the framework.

Another problem for decision-making generally is 'reflexivity', where the plan may affect the situation. For example, where we are planning repeated convoys along a route, so that the existing traffic may get displaced. We may be looking to judge the capacity of nearby roads to absorb displaced traffic, and hence how much traffic is left to hold up our convoys. Again, this makes the 'outcome' difficult to judge, and invalidates the simple notion of 'expected delay', but does not invalidate the framework. We can still judge the 'uiltity' of a plan in a given situation. Thus the extension proposed to planning practice does not preclude other desirable extensions.

Conclusions

AI Planning is relatively straightforward when dealing with a simple 'puzzle' world, where the main challenge is computational cost. Uncertainty introduces a new dimension, which has been widely studied from a Bayesian point of view, but with limited success. This paper has described a conceptual framework for uncertainty in planning that re-uses the formalism of game-theory to situations where the source of uncertainty, such as the weather, is not 'against us'. A basic method, a kind of search strategy, has been described to validate the feasibility of using the framework on practical planning problems subject to the type of uncertainty described. Some opportunities for domain-specific tuning have also been indicated.

Simple examples relating to route-planning have been described, showing how this type of uncertainty arises naturally in route planning problems, and is not adequately addressed by 'Bayesian planning'. For the simple examples it is straightforward and, I claim, natural to solve the corresponding planning problem. For more complex problems, such as those where the power of AI planning tools is required, a general heuristic has been described for leveraging existing planning practises and tools, provided that they meet some simple and intuitive criteria of configurability.

Bayesian planning is often not practical. The proposed method allows one to take plans from existing planners and select the best. It is more natural and practical than Bayesian planning. A Bayesian planner, if available, would provide a good 'reference' plan, but, as the examples have shown, does not always provide a very satisfactory plan.

It has been argued that the use of this framework is consistent with other ways of extending conventional planning to overcome recognised deficiencies. The main problem with the method is in generating good 'compromise' plans. But what makes a good compromise is inevitably very domain-dependent, and so no planning method could satisfactorily handle real-world uncertainties for all types of planning problem.

The new approach described here can be used to evaluate plans against possible scenarios, to select the more robust plans. It will not do worse than the conventional approach. However, more work needs to be done to understand the factors that contribute to non-Bayesian uncertainty. Doing better than the conventional approach may not be enough. Specific domains need to be studied, to see if the plan generation method described here is satisfactory. The most important work required is to find reliable methods of identifying problem domains, and situations within domains, where the deficiencies of the 'Bayesian' approach are significant and the new approach will remedy or ameliorate them.

References

Bayes T. 1763 An Essay Towards Solving a Problem in the Doctrine of Chances *Philosophical Transactions* Essay LII

Blythe J. 1999. Decision-theoretic Planning *AI Magazine* Vol. 20 No. 2

Bernstein P 1996 *Against the Gods* Wiley

Cheeseman P.1985 *In Defense of Probability* Proceedings of the Ninth International Joint Conference on Artificial Intelligence, Morgan Kaufmann

D'Ambroise B. 1999. Inference in Bayesian Networks *AI Magazine* Vol. 20 No. 2

Doyle, J. and Thomason, R.H. 1999. Background to Qualitative Decision Theory *AI Magazine* Vol. 20 No. 2

Enthoven, A.C. and Smith, K.W. 1971 *How Much is Enough? Shaping the Defense Program 1961-1969* Harper & Row

Haddawy, P. 1999. An Overview of Some Recent Developments in Bayesian Problem-Solving Techniques *AI Magazine* Vol. 20 No. 2

Keynes J.M. 1921 *A treatise on probability* London

Russell.S & P. Novig.P (Eds) 1995 *Artificial Intelligence: a modern approach* Prentice Hall

Smith A. 1776 *Wealth of Nations* Various

Smith C.A.B. 1961 *Consistency in statistical inference and decision* Journal of the Royal Statistical Society, Series B, 23 1-37

Soros G. 1996 *A Failed Philosopher Tries Again* The critical rationalist Vol. 01 No. 01 26 Nov. 1996

Skyrms B. 2000. Rationality and Evolution of the Social Contract *Journal of Consciousness Studies* Vol. 7, No. 1/2

Von Neumann, J. and Morgenstern, O. 1944. *Theory of Games and Economic Behaviour* Princeton University Press

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

Acknowledgements

This work has been carried out as part of Technology Group 10 (Computing, Information and Signal Processing) of the UK Ministry of Defence Corporate Research Programme. I would also like to thank colleagues at DERA and AIAI (University of Edinburgh) who have debated the issues of uncertainty in both decision-making generally and planning, and helped me to isolate the relatively accessible aspects covered by this paper.