Decision Aiding and Argumentation
(preliminary draft)

Pavlos Moraïtis(+) and Alexis Tsoukiàs(*)
(+)Computer Science Department, University of Cyprus
e-mail: moraitis@ucy.ac.cy
(*) LAMSADE-CNRS, Université Paris Dauphine
e-mail: tsoukias@lamsade.dauphine.fr

Abstract

In this paper we show how argumentation theory can be used in order to model the decision aiding process, besides being a formalism enabling to take in account the defeasible character of the outcomes of such a process. Under such a perspective we show how autonomous agents can be enabled in decision support activities with respect to a user.

1 Introduction

The aim of this paper is, on the one hand to introduce elements of argumentation theory in decision aiding methodology (how to revise the cognitive artifacts of a decision aiding process) and on the other hand to show the usefulness of using decision theoretic concepts in designing autonomous agents with decision support capabilities.

The paper is structures as follows. In section 2 we briefly introduce the concept of decision aiding process, the cognitive artifacts it produces and the methodological problems which such a presentation opens. An example will be used for this purpose. In section 3 we briefly introduce argumentation theory. In section 4 we show how this theory can be used in conducting a decision aiding process and therefore how it can be used for enhancing decision support capabilities of autonomous agents. In section 5 we discuss how such findings can improve both decision aiding methodology and agent theory.

2 The Decision Aiding Process

Decision Aiding is an activity occurring in the everyday life of almost everybody. In this paper we are interested to that particular type of decision aiding where formal and abstract languages are used (different decision theories and approaches). Although decision aiding is a distributed process of cognition we will present this concept using an operational approach based on the identification of the cognitive artifacts of the process (the outcomes or deliverables). For more details the reader can see [2, 8, 9].

2.1 Cognitive Artifacts

A decision aiding process is a decision process involving at least two actors: a client, who himself is involved to at least one decision process (the one generating the concern for which the aid is requested) and the analyst who is expected to provide the decision support. The aim of this particular process is to establish a shared representation of the client's concern, using the analyst's methodological knowledge, a representation enabling to undertake an action towards the concern. The outcomes of this process are:

- A representation of the problem situation.

  The construction of such an artifact allows, on the one hand the client to better understand his position within the decision process for which he asked the decision support and on the other hand the analyst to better understand his role within this decision process. From a formal point of view a representation of the problem situation is a triplet: $\mathcal{P} = \langle \mathcal{A}, \mathcal{O}, \mathcal{S} \rangle$ where:
  - $\mathcal{A}$ is the set of participants to the decision process;
  - $\mathcal{O}$ is the set of stakes each participant brings within the decision process;
  - $\mathcal{S}$ is the set of resources the participants commit on their stakes and the other participants' stakes.

- The establishment of a problem formulation.

  For a given representation of the problem situation the analyst might propose to the client one or more "problem formulations". This is a crucial point of the decision aiding process. The representation of the problem situation has a descriptive (at the best explicative objective). The construction of the problem formulation introduces what we call a model of rationality. A problem formulation reduces the reality of the decision process within which the client is involved to a formal and abstract problem. The result is that one or more of the client's concerns are transformed to formal problems on which we can apply a method (already existing, adapted from an existing one or created ad-hoc) of the type studied in decision theory. From a formal point of view a problem formulation is a triplet: $\Gamma = \langle A, V, \Pi \rangle$ where:
  - A: is the set of potential actions the client may undertake within the problem situation as represented in $\mathcal{P}$;

- V: is the set of points of view under which the potential actions are expected to be observed, analysed, evaluated, compared etc., including different scenarios for the future;
- Π: is the problem statement, the type of application to perform on the set A, an anticipation of what the client expects (for instance Π can be a choice statement or a ranking or a classification or a reject, for details see [1],[9]).

- The construction of an evaluation model.

  With this term we indicate what traditionally are the decision aiding models conceived through any operational research, decision theory or artificial intelligence method. Classic decision aiding approaches will focus their attention on the construction of this model and consider the problem formulation as given. An evaluation model is an n-uplet: $\mathcal{M} = \langle A^*, D, G, \Omega, \mathcal{R} \rangle$ where:
  - A^* a set of alternatives on which the model will apply;
  - D is a set of dimensions (attributes) under which the elements of A are observed, measured, described etc. (such a set can be structured, for instance through the definition of an hierarchy); a scale is associated to each element of D;
  - G is a set of criteria (if any) under which each element of A^* is evaluated in order to take in account the client's preferences;
  - Ω is an uncertainty structure;
  - $\mathcal{R}$ is a set of operators enabling to obtain synthetic information about the elements of A or of $A \times A$, namely aggregation operators (of preferences, of measures, of uncertainties etc.).

- The establishment of a final recommendation.

  The final recommendation represents the return to reality for the decision aiding process. Usually the evaluation model will produce a result, let's call it Φ. The final recommendation should translate such a result from the abstract and formal language in which Φ is formulated to the current language of the client and the decision process where he is involved. Some elements are very important in constructing this artifact:
  - the analyst has to be sure that the model is formally correct;
  - the client has to be sure that the model represents him, that he understands it and that he should be able to use its conclusions (the client should feel "owner"'of the results, besides being satisfied of them);
  - the recommendation should be "legitimated" with respect to the decision process for which the decision aiding has been asked.

2.2 Conducting the process

Conducting a decision aiding process is not a linear process where we establish the four cognitive artefacts one after the other. Since a decision aiding process always refers to a decision process which has a time and space extension it is natural that the outcomes of the decision aiding process remain *defeasible cognitive artefacts*. Usually the process will encounter situations where any of the above artefacts:
- may conflict with the evolution of the client's expectations, preferences and knowledge;
- may conflict with the updated state of the decision process and the new information available.
It is therefore necessary to adapt the contents of such artefacts as the decision aiding process evolves in time and space. Consider the following decision process.

Example 2.1. *An agent wants to make something this evening. (S)he might see a friend or go outside alone or stay at home. In the last case the options are either to watch TV or to go to bed. If (s)he goes outside alone, the options are either to go to a restaurant (and then a restaurant has to be identified) or to a concert (and then a concert has to be identified) or to go to the bar in front of the house. If (s)he is going to meet a friend then, potentially, there are Helen, Mary and Bob and they could meet either to go to a cinema (and then a movie has to be identified) or to go to disco (then a disco has to be identified) or meet somewhere else (where?). The agent decided that prefers to meet a friend. But then Helen is not available, Mary does not like neither cinema nor disco (she wants to come to the agent's home), while Bob is available for everything. The agent at this point decides that these solutions are not really satisfactory and that (s)he prefers to go outside alone. However, now there is no more time to find a restaurant or a concert and therefore decides to go to the bar in front of the house.*

Suppose that for the different decisions of this agent a decision support is asked. In such a situation the descriptive model of the decision aiding process can turn to be useful since it allows to fix which elements have to be modified and why. Coming back to the example we may consider it as a planning problem and call the possible plans the agent may execute as π_j, each of them composed by sequences of actions $\alpha_i(x_1, \cdots x_n)$ where the x represent variables which can can be instantiated within definite domains. The problem could then be described as follows ('|' represents execution of two actions contemporaneously, while ';' represents execution of two actions sequentially):
π_0: `do_something_this_evening`(x_1)
$x_1 \in \{\pi_1, \pi_2, \pi_3\}$ where
π_1: `(find_a_friend(`x_2`)`$|$`find_a_place(`x_3`))`$;$`see_friend(`x_2, x_3`)`

π_2: `go_outside(`x_4`)`
π_3: `stay_home(`x_5`)`
$x_2 \in F = \{$`Helen,Mary,Bob`$\}$
$x_3 \in L = \{$`Cinema,Disco,Other`$\}$
$x_4 \in \{\pi_4, \pi_5, \pi_6\}$ where
π_4: `(find_restaurant(`x_6`);go_restaurant(`x_6`))`
π_5: `go_to_bar`
π_6: `(find_concert(`x_7`);go_concert(`x_7`))`
$x_5 \in \{$`TV,bed`$\}$
$x_6 \in \{r_1, \cdots r_n\}$
$x_7 \in \{c_1, \cdots c_n\}$.

The problem formulation in this case will be $\Gamma_1 = \langle A_1, V_1, \Pi_1 \rangle$ where A_1 is the set of any possible plan among the above not necessarily instantiated with precise values for the variables, V_1 includes pleasure and friends availability and Π_1 is a choice (of the best thing to do this evening) statement. An alternative problem formulation could be $\Gamma_2 = \langle A_2, V_2, \Pi_2 \rangle$ where A_2 is like A_1, V_2 concerns time availability and Π_2 is always a choice statement. However, Γ_2 will occur only exceptionally (when time becomes a critical issue).

The intuitive reasoning of the above agent could be: usually friends are always available, therefore usually this an irrelevant dimension for my choice. However, in the case friends are not available then "with who I may meet" might become a priority. The above idea corresponds to the establishment of two evaluation models where $\mathcal{M}_1$ is "preferred" to $\mathcal{M}_2$:

- $\mathcal{M}_1 : \langle\ A = \{\pi_1, \pi_2, \pi_3\}$,
  $D =$ pleasure,
  $G = \{$pleasure : $\pi_1 >_p \pi_2 >_p \pi_3\}$,
  $\Omega = \emptyset$,
  $\mathcal{R} =$ choice)$\rangle$
- $\mathcal{M}_2 : \langle\ A = \{\pi_1(x_2, x_3) : (x_2, x_3) \in F \times L\}$,
  $D =$ attractiveness, the associate scale being $\{$Interesting,Acceptable,Non-interesting$\}$
  $G = \{$pleasure :I>A>N,
  $\Omega = \emptyset$,
  $\mathcal{R} =$ (classification, $\rangle$

Applying $\mathcal{M}_1$ the agent will choose π_1. In implementing this solution there are potentially 9 alternatives, but the result of the inquiry leads to only four:

$\{$(Bob,Cinema;BC),(Bob,Disco;BD),(Bob,Other;BO),(Mary,Other;MO)$\}$. The agent therefore has to apply $\mathcal{M}_2$. For this purpose (s)he uses the following model:
- all alternatives involving Helen are N;
- all alternatives involving Mary and Cinema or Disco are I;
- all other alternatives are NI.
All solutions are unsatisfactory and the solution set is empty.

The agent has to reconsider his problem and will be oriented in implementing his second best option: π_2. However, meanwhile the time dimension became critical. Therefore problem formulation Γ_2 becomes relevant, leading to the following evaluation model:
$\mathcal{M}_3 : \langle\ A = \{\pi_4, \pi_5, \pi_6\}$,
$D =$ (time forecast for each plan,
$t(\pi_4) = 1000, t(\pi_5) = 10, t(\pi_6) = 1000)$,
$G = \{$time : $x >_t y$ **iff** $t(x) < t(y)\}$,
$\Omega = \emptyset$,
$\mathcal{R} =$ choice)$\rangle$
and the choice will be plan π_5 which finally is implemented.

2.3 Where is the problem?

The above example shows that during a decision aiding process several different versions of the cognitive artifacts are established. However, such different versions are strongly related among them since they carry essentially the same information and only small part of the model has to be revised. The problem is: is it possible to give a formal representation of how such an evolution occurs? In other terms: is it possible to show how a set of alternatives or some preferential information may change while shifting from one model to another? The reasons for such an analysis are the following:
- on the one hand we would like to design autonomous agents able to undertake decision aiding tasks (which, by the way, is what in reality most of the agents do with respect to their users);
- on the other hand we want to show why such a theory could be helpful for automatic decision purposes in autonomous agents.
For this purpose we use argumentation theory as a first step in integrating decision aiding methodology and formal reasoning languages.

3 Argumentation Theory

Autonomous agents need to make decisions under complex preferences policies that take into account different factors. These policies have a dynamic nature and are influenced by the particular state of the environment in which the agent finds himself. In this section we will present the basic concepts of the argumentation framework we will use and which is originally presented in (see [5, 6]) In this framework it is assumed that agents are always associated with an (social) environment of interaction. This context is called context of interaction (i.e. market, enterprise, etc.). This determines the relationship between the possible roles the different agents can have within this environment and can influence the decision policies used by the agents. Generally, roles are associated to a default context that defines shared social relations of different forms (e.g. authority, friendship, relationship, etc.) and specifies the interaction between different roles. For instance in the army context an officer gives orders that are obeyed by a soldier, or in the market a regular customer has benefits over a normal customer. Consequently, a default context can be understood via priorities between the different roles. However, a default context determining the basic roles filled by the agents is not the only environment where they could interact. For example two friends can also be colleagues, an officer and a soldier can be family friends in civil life. Therefore we consider a second level of context, called specific context, which can overturn the pre-imposed, by the default context, ordering between roles and establish a different relation between them. For instance, the authority relationship between an officer and a soldier would change under the specific context of a social meeting or regular and normal customers may have an equivalent consideration during the high season.

The argumentation framework that we will use was originally proposed in [4] and extended further in [5],[6] in order to deal with the dynamic nature of interaction of autonomous agents. It uses a priority relation between arguments in the theory which is not a static relation but context dependent that captures the non-static preferences associated to roles and specific context on interaction. The approach follows the same line of [3, 7], where this dynamic priority relation is defined as part of the agent's theory with the same argumentation semantics along with the rest of the theory.

An (extended) argumentation theory is then defined as follows.

Definition 3.1. *A theory is a pair* $(\mathcal{T}, \mathcal{P})$ *whose sentences are formulae in the* **background monotonic logic** $(\mathcal{L}, \vdash)$ *of the framework defined as* $L \leftarrow L_1, \ldots, L_n$*, where* $L, L_1, \ldots, L_n$ *are positive or explicit negative ground literals. For rules in* $\mathcal{P}$ *the head* L *refers to an (irreflexive) higher-priority relation, i.e.* L *has the gen-*

eral form $L = h\_p(rule1, rule2)$. The derivability relation, $\vdash$, of the background logic is given by the single inference rule of modus ponens.

For simplicity, we will assume that the conditions of any rule in the theory do not refer to the predicate h_p thus avoiding self-reference problems. For any ground atom $h\_p(rule1, rule2)$ its negation is denoted by $h\_p(rule2, rule1)$ and vice-versa.

An **argument** for a literal L in a theory $(\mathcal{T}, \mathcal{P})$ is any subset, T, of this theory that derives L, $T \vdash L$, under the background logic. In general, we can separate out a part of the theory $\mathcal{T}_0 \subset \mathcal{T}$ and consider this as a non-defeasible part from which any argument rule can draw information that it might need. The notion of attack between arguments in a theory is based on the possible conflicts between a literal L and its negation and on the priority relation given by h_p in the theory.

Definition 3.2. *Let $(\mathcal{T}, \mathcal{P})$ be a theory, $T, T' \subseteq \mathcal{T}$ and $P, P' \subseteq \mathcal{P}$. Then (T', P')* **attacks** *(T, P) iff there exists a literal L, $T_1 \subseteq T'$, $T_2 \subseteq T$, $P_1 \subseteq P'$ and $P_2 \subseteq P$ s.t.:*

(i) $T_1 \cup P_1 \vdash_{min} L$ *and* $T_2 \cup P_2 \vdash_{min} \neg L$

(ii) $(\exists r' \in T_1 \cup P_1, r \in T_2 \cup P_2$ *s.t.* $T \cup P \vdash h\_p(r, r')) \rightarrow (\exists r' \in T_1 \cup P_1, r \in T_2 \cup P_2$ *s.t.* $T' \cup P' \vdash h\_p(r', r))$.

Here, when L does not refer to h_p, $T \cup P \vdash_{min} L$ means that $T \vdash_{min} L$. This definition states that a (composite) argument (T', P') attacks (or is a counter-argument to) another such argument when they derive a contrary conclusion, L, and $(T' \cup P')$ makes the rules of its counter proof at least "as strong" as the rules for the proof by the argument that is under attack. Note that the attack can occur on a contrary conclusion L that refers to the priority between rules.

Definition 3.3. *Let $(\mathcal{T}, \mathcal{P})$ be a theory, $T \subseteq \mathcal{T}$ and $P \subseteq \mathcal{P}$. Then (T, P) is* **admissible** *iff $(T \cup P)$ is consistent and for any (T', P') if (T', P') attacks (T, P) then (T, P) attacks (T', P'). Given a ground literal L then L is a* **credulous (resp. skeptical) consequence** *of the theory iff L holds in a (resp. every) maximal (wrt set inclusion) admissible subset of $\mathcal{T}$.*

Hence when we have dynamic priorities, for an object-level argument (from $\mathcal{T}$) to be admissible it needs to take along with it priority arguments (from $\mathcal{P}$) to make itself at least "as strong" as the opposing counter-arguments. This need for priority rules can repeat itself when the initially chosen ones can themselves be attacked by opposing priority rules and again we would need to make now the priority rules themselves at least as strong as their opposing ones.

We can now define an agent's argumentation theory for describing his policy in an environment with roles and context as follows.

Definition 3.4. *An agent's* **argumentative policy theory or theory,** T**,** *is a triple* $T = (\mathcal{T}, \mathcal{P}_R, \mathcal{P}_C)$ *where the rules in* $\mathcal{T}$ *do not refer to* h_p*, all the rules in* $\mathcal{P}_R$ *are priority rules with head* $h\_p(r_1, r_2)$ *s.t.* $r_1, r_2 \in \mathcal{T}$ *and all rules in* $\mathcal{P}_C$ *are priority rules with head* $h\_p(R_1, R_2)$ *s.t.* $R_1, R_2 \in \mathcal{P}_R \cup \mathcal{P}_C$*.*

We therefore have three levels in an agent's theory. In the first level we have the rules $\mathcal{T}$ that refer directly to the subject domain of the agent. We call these the **Object-level Decision Rules** of the agent. In the other two levels we have rules that relate to the policy under which the agent uses his object-level decision rules according to roles and context. We call the rules in $\mathcal{P}_R$ and $\mathcal{P}_C$, **Role (or Default Context) Priorities** and **(Specific) Context Priorities** respectively.

In a more general setting, we can associate the dynamic preferences expressed through the rules of the two later levels, to normal situations (related to a default context) and specific situations (related to specific contexts).

In several cases the admissibility of an argument depends on whether we have or not some background information about the specific case in which we are reasoning. Sometime this information maybe just unknown and agents can reason further to find assumptions related to the unknown information under which they can build an admissible argument. We can formalise this conditional form of argumentative reasoning by defining the notion of supporting information and extending argumentation with abduction on this missing information.

We can formalise this conditional form of argumentative reasoning by defining the notion of *supporting information* and extending argumentation with *abduction* on this missing information.

Definition 3.5. *Let* $T = ((\mathcal{T}_0, \mathcal{T}), \mathcal{P})$ *be a theory, and* $\mathcal{A}$ *a distinguished set of predicates in the language of the theory, called* **abducible** *predicates*[1]*. Given a goal* G*, a set* S *of abducible literals consistent with the non-defeasible part* $\mathcal{T}_0$ *of* T*, is called a* **strong (resp. weak) supporting information** *for* G *iff* G *is a skeptical (resp. credulous) consequence of* $(\mathcal{T}_0 \cup S, \mathcal{T}, \mathcal{P})$*.*

The structure of an argument can also be generalised as follows.

Definition 3.6. *Let* $T = ((\mathcal{T}_0, \mathcal{T}), \mathcal{P})$ *be a theory and* $\mathcal{A}$ *its abducible predicates. A* **supported argument** *in* T *is a tuple* (Δ, S)*, where* S *is a set of abducible literals consistent with* $\mathcal{T}_0$ *and* Δ *is a set of argument rules in* T*, which is not admissible in* $((\mathcal{T}_0, \mathcal{T}), \mathcal{P})$*, but is admissible in the theory* $((\mathcal{T}_0 \cup S), \mathcal{T}, \mathcal{P})$*. We say that* S *supports the argument* Δ *and that* Δ *is conditional on* S*.*

[1] Typically, the theory $\mathcal{T}$ does not contain any rules for the abducible predicates.

The supporting information expressed through the abducible predicates refers to the incomplete and evolving information of the external environment of interaction. Typically, this information pertains to the context of the environment, the roles between agents or any other aspect of the environment that is dynamic.

4 Argumentation in Decision Aiding

Let's go back to the example of section 2. We will focus our attention now on the interaction between problem formulation and evaluation model and how these can evolve during the decision aiding process. The following is of course a simplified representation of the process aiming just to show how argumentation is used.

We first establish a number of object level rules showing the relations between problem formulations and evaluation models. As far as the example is concerned we describe the fact that problem formulation Γ_1 can entail two evaluation models: $\mathcal{M}_1, \mathcal{M}_2$ and that problem formulation Γ_2 entails the evaluation model $\mathcal{M}_3$. Moreover it is shown that when $\mathcal{M}_1$ is applied then usually we have to continue applying $\mathcal{M}_2$ (in order to choose the friends to meet). Finally, in the case friends are not available (NFA) then model $\mathcal{M}_1$ should usually apply in order to choose another feasible solution. It should be noted that applying a model automatically implies that other models do not apply:
- rule r_1: $\mathcal{M}_1 \leftarrow \Gamma_1$
- rule r_2: $\phi_1 \leftarrow \mathcal{M}_1$
- rule r_3: $\mathcal{M}_2 \leftarrow \phi_1$
- rule r_4: $\mathcal{M}_2 \leftarrow \Gamma_1$
- rule r_5: $\mathcal{M}_3 \leftarrow \Gamma_2$
- rule r_6: $\mathcal{M}'_1 \leftarrow \Gamma_1, NFA$

We can now establish the default context priority rules which help in applying the object level ones. In the case of the example, in the default context priority is given in applying evaluation model $\mathcal{M}_1$ unless we occur in the case where we already know who are the available friends (FA) or time is missing (NTA). In the first case model $\mathcal{M}_2$ should apply, while in the second case problem formulation Γ_2 holds:
- rule R_1: $h\_p(r_1, r_4)$
- rule R_2: $h\_p(r_1, r_5)$
- rule R_3: $h\_p(r_4, r_1) \leftarrow FA$
- rule R_4: $h\_p(r_5, r_1) \leftarrow NTA$
- rule R_5: $h\_p(r_6, r_1)$

Finally we write down the specific context rules which will give priority to the exceptional conditions rules, in the case such exceptions indeed hold:
- rule C_1: $h\_p(R_3, R_1)$
- rule C_2: $h\_p(R_4, R_2)$

Applying the above theory what we get is that usually we apply model $\mathcal{M}_1$ and then model $\mathcal{M}_2$ unless friends are already available which implies applying $\mathcal{M}_2$ directly. Exceptionally (time missing) we apply model $\mathcal{M}_3$.

5 Discussion

The previous example roughly shows how is possible to simulate the reasoning of an agent facing an apparently simple problem which confronted with the reality of the decision process becomes more and more complex since the outcomes of the decision aiding process have to evolve. What do we obtain?

On the one hand the use of the cognitive artefacts of the decision aiding process enables to focus on the critical elements for the final recommendation (do we have to modify the set of alternatives? the set of criteria? the problem formulation altogether?). On the other hand the use of argumentation theory enables to establish a formal reasoning sufficiently flexible to account for the dynamics of the decision process and the defeasible character of the decision aiding process outcomes.

What do autonomous agents do with respect to their users? Our claim is that they provide decision support to different levels: from simply creating meaningful information up to substituting the user in some decision task. It is clear that agents endowed with the possibility to control the cognitive artefacts of the decision aiding process in which they are engaged are advantaged since they can focus on the critical elements of such artefacts. Moreover, agents with argumentation capabilities are also able to handle the dynamic nature of the decision process and to adapt their recommendations.

6 Conclusions

In the paper we show how argumentation can help in improving the decision aiding process enabling to describe and control the evolution of the cognitive artefacts of this process. The outcomes of the decision aiding process are described in the paper through an operational model. Then argumentation theory is introduced and we show how can be used in order to take in account the defeasible character of the decision aiding process outcomes. Future research is expected to further investigate the relation between reasoning, argumentation and decision aiding including learning and possibly other forms of non monotonic reasoning.

References

[1] C.A. Bana e Costa. Les problematiques de l'aide a la decision: vers l'enrichissement de la trilogie choix-tri-rangement. *RAIRO/ Recherche Operationnelle*, 30(2):191–216, 1996.

[2] D. Bouyssou, T. Marchant, M. Pirlot, P. Perny, A. Tsoukiàs, and Ph. Vincke. *Evaluation and decision models: a critical perspective*. Kluwer Academic, Dordrecht, 2000.

[3] G. Brewka. Well founded semantics for extended logic programs with dynamic preferences. *Journal of Artificial Intelligence Research*, 4:19–36, 1996.

[4] A.C. Kakas, P. Mancarella, and P.M. Dung. The acceptability semantics for logic programs. In *Proceedings of ICLP94*, pages 504–519, 1994.

[5] A.C. Kakas and P.Moraïtis. Argumentative agent deliberation, roles and context. In J. Dix, J. A. Leite, and K. Satoh, editors, *Electronic Notes in Theoretical Computer Science*, volume 70(5). Elsevier, Amsterdam, 2002. 15 pages.

[6] A.C. Kakas and P.Moraïtis. Argumentation based decision making for autonomous agents. In *Proceedings of AAMAS03*, pages 883–890. ACM Press, New York, 2003.

[7] H. Prakken and G. Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artificial Intelligence and Law*, 4:331–368, 1996.

[8] A. Tsoukiàs. From decision theory to decision aiding methodology. Technical Report 2003-21, DIMACS, Rutgers University, 2003. to appear with the European Journal of Operational Research.

[9] A. Tsoukiàs. On the concept of decision aiding process. Technical Report 2003-??, DIMACS, Rutgers University, 2003.