Argumentation Based Modeling of Decision Aiding for Autonomous Agents

Yannis Dimopoulos
Dept. of Computer Science, University of Cyprus
yannis@cs.ucy.ac.cy

Pavlos Moraitis
Dept. of Computer Science, University of Cyprus
moraitis@cs.ucy.ac.cy

Alexis Tsoukiàs
LAMSADE-CNRS, Université Paris Dauphine
tsoukias@lamsade.dauphine.fr

Abstract

Decision Aiding can be abstractly described as the process of assisting a user/client/decision maker by recommending possible courses of his action. This process has to be able to cope with incomplete and/or inconsistent information for the following reasons. First, the recommendations provided during this process depend heavily on the environment the decision is made. Since complete knowledge of this environment is almost impossible, decision aiding has to be carried out under incomplete information. Second, the decision aiding process also depends on the preferences of its user. However, such subjective information is affected by uncertainty, possible inconsistencies and is dynamically revised due to the time dimension of decision aiding. A complete description of a model of the user is also almost impossible, therefore a decision aiding process must also account for this source of incompleteness. This paper presents a model of Decision Aiding that is amenable to automation and shows how it can be embedded in autonomous agents and thus give them the capability to provide decision aiding to a human user or to completely substitute him for some decision task. The whole process is modelled in an suitable argumentation framework that treats Decision Aiding as an iterative defeasible reasoning process.

1. Introduction

Decision Analysis ([3], [4], [5]) is concerned with the process of providing decision support to "clients" that feel unable to handle alone a problem situation. We call such an activity "decision aiding". Decision Aiding is a process characterised by the emergence of cognitive artifacts, resulting from the interaction between the "client" and the "analyst". The decision analyst and the client are engaged in an iterative process, where the analyst attempts, through successive steps of interaction with the client, to obtain a better understanding of the problem the client is facing. To be able to cope with the complexity of both the real world and the needs of the client, the analyst needs to make assumptions and reason as if these assumptions were true in the world. The recommendations, that are the outcomes of the Decision Aiding process, are subject to the client validation. Rejection of the recommendations means that some of the assumptions made by the analyst are false and must be retracted. In this case, the negation of these assumptions is added to the theory representing the knowledge of the agent. The revision of the assumptions triggers the revision of the cognitive artifacts generated during the previous Decision Aiding cycle. This iterative process continues until the recommendations satisfy the user.

This paper presents a model of Decision Aiding that is amenable to automation. It also aims to show the usefulness of using such decision theoretic concepts in the design of autonomous agents with decision support capabilities. In this model, Decision Aiding is understood as the process of constructing and revising cognitive artifacts, that gradually transform an abstract problem description to an invocation of a concrete decision support tool. Therefore, automating the Decision Aiding process amounts to automating the construction of these cognitive artifacts taking also into account the defeasible character of the process.

Our aim is to endow agents with such decision aiding capabilities so that they can provide decision aiding on different levels: from simply creating meaningful recommendations up to substituting the user in some decision task.

Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04)
0-7695-2101-0/04 $ 20.00 IEEE

Agents endowed with the possibility to control the cognitive artifacts of the decision aiding process in which they are engaged are advantaged. The reason is that they can focus on the critical elements of such artifacts, and therefore easily detect the parameters of the decision that must be updated in order to adapt this decision (e.g. best choice, ranking or classification of possible alternatives) to the specific circumstances.

As the construction of the cognitive artifacts is based on retractable assumptions, the formal language to be used for the process modelling must be a non-monotonic one. In the last few years there has been a growth of interest in the use of argumentation [12] in the multi-agent field. It has been shown that argumentation is very well suited for agent self-deliberation [8], negotiation [11] or communication [2, 7]. In this work we show that argumentation is also well suited for modelling a decision aiding process. For this reason we use an argumentation framework [8] which is dynamic, with arguments and their strength depending on the particular context that the agent finds himself, thus allowing the agent to adapt his decisions in a changing environment. The Decision Aiding theories can be easily implemented directly from their declarative specification in the Gorgias system [18] for this framework.

The approach presented in this paper is the first attempt to introduce a formalism that describes the decision aiding process not only for descriptive purposes, but also for prescriptive. The Decision Analysis literature (see e.g. [3], [10], [13], [17]) is mainly focussed on the interactions between humans (mainly the clients and the analysts). There is very little attention, if any, to the use of decision theories and decision aiding methodology when this interaction is between a human (a user) and an automatic device (such as an autonomous agent). Therefore, existing methods that have been proposed to support the decision aiding process and the construction of its cognitive artifacts, are not expected to be formalized, thus missing any tentative to establish a formal theory for them. To the best of our knowledge, this approach is an original attempt in the multi-agent field on the use of argumentation for the modelling of an automated decision aiding model for autonomous agents.

The paper is structured as follows. In section 2 we briefly introduce the concept of decision aiding process and the cognitive artifacts it produces. In section 3 we briefly introduce the argumentation framework we will use. In section 4 we show how this framework 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 present conclusions and future work.

2. The Decision Aiding Process

Decision Aiding is an activity occurring in the everyday life of almost everybody. In this paper we are interested in that particular type of decision aiding where formal and abstract languages are used (different decision theories and approaches). A decision aiding process is a particular type of decision process involving at least two actors: a client, who himself is involved in 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.

2.1. Cognitive Artifacts

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 is referred to [4, 15, 16]. The outcomes of this process are:
- a representation of the problem situation: $\mathcal{P}$;
- the establishment of a problem formulation: Γ;
- the construction of an evaluation model: $\mathcal{M}$;
- the establishment of a final recommendation: Φ.

In this paper we will focus on the establishment of Γ and the construction of $\mathcal{M}$. Our interest in this part of the process is due to the fact that both such artifacts represent the more formalised and structured outcomes of the process. Therefore, they allow to be easily modelled using a formal language. Two points should be considered:
- although these two artifacts appear subsequent they are constructed through continuous interactions;
- the way the decision aiding process is conducted influences the process outcomes.

We can now go through more details as far as these two artifacts are concerned.

Problem formulation (Γ): 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) purpose. 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 $\mathcal{P}$. It should be noted that these are not "given", but have to be constructed. A typical situation is the refinement of abstract options to more precise actions. A does not necessarily have a formal structure.
- 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. A problem statement can be operational or not (such as describing or conceiving the elements of A). Operational problem statements are partitioning operations to be applied on the set α within the evaluation model $\mathcal{M}$. As such they can partition the set α:
- in predefined categories (large-medium-small, illness(A)-illness(B)) or in categories to be inferred comparing the elements of α (clusters, choice sets);
- in ordered categories (accepted-rejected, bad-medium-good) or unordered categories (greens-blacks, monkeys-elephants).
Therefore, Π can be a choice statement, a ranking, a classification, a reject, a clustering etc. (for details see [1], [16]).

Evaluation Model ($\mathcal{M}$): 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 a tuplet: $\mathcal{M} = \langle \alpha, D, G, \Omega, \mathcal{R} \rangle$ where:
- α is a precise set of alternatives or decision variables on which the model will apply; α has a precise structure: enumeration of actions, domain of real numbers, combinatorial structure etc.;
- D is a set of dimensions (attributes) under which the elements of A are observed, measured, described etc.; a scale is always associated to each element of D;
- G is a set of criteria (if any) under which each element of α is evaluated in order to take into 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.).

All that said, the question to which we focus our attention in the paper is: how the establishment of a problem formulation influences the construction of an evaluation model and vice versa.

2.2. Conducting the process

Conducting a decision aiding process is not a linear process where we establish the four cognitive artifacts one after the other. Since a decision aiding process always refers to a decision process which has a time extension and is situated with respect to a specific context, it is natural that the outcomes of the decision aiding process remain *defeasible cognitive artifacts*. Usually the process will encounter situations where any of the above artifacts:
- 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 artifacts as the decision aiding process evolves in time and within a specific context.

The recommendation Φ is submitted to the user. Three possibilities hold:
Φ_1: the recommendation is validated and implementable;
Φ_2: the recommendation is validated, but fails to be implemented;
Φ_3: the recommendation is not validated;
The way the recommendation is submitted to the user is out of the scope of this paper.

The agent therefore receives a pair $\langle \Phi_j, T \rangle$ where:
- Φ_j represents the state of the recommendation with the user;
- T represent the reasons for which the recommendation is in such a state.
In case the recommendation is in state 1 then the reasons in T are the overall appreciation of the user. In case the recommendation is in state 2 or 3 such reasons can be:
- no feasible solution in α satisfies the user or the recommendation is no more feasible;
- the available measures of elements in α are considered irrelevant, erroneous or affected by too large uncertainty;
- the preference models applied on α are not reliable and the user feels not to be correctly represented;
- uncertainty is mis-represented;
- the aggregation procedure is revealed to be meaningless or irrelevant to the user.

3. The Argumentation Framework

This section gives the basic concepts of the underlying argumentation framework in which an agent represents and reasons with its decision aiding theory. This framework was proposed in [6] and developed further in [8], in order to accommodate a dynamic notion of priority over the rules (and hence the arguments) of a given theory.

Furthermore, we will see that (components of) an agent's theory will be layered in three levels. *Object-level decision rules*, are defined at the first level. The next two levels, describe priority rules on the decision rules of the first level and on themselves thus expressing a preference policy for the overall decision making of the agent. This policy is separated into two levels: level two to capture the *default* preference policy under normal circumstances while level three is concerned with the *exceptional* part of the policy that applies under specific contexts. The argumentation-based decision making will then be sensitive to context changes.

In general, an argumentation theory is defined as follows.

Definition 3.1. *A theory is a pair $(\mathcal{T}, \mathcal{P})$. The sentences in $\mathcal{T}$ are propositional 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. Rules in $\mathcal{P}$ are the same as in $\mathcal{T}$ apart from the fact that the head L of the rules has the general form $L = h\_p(rule1, rule2)$ where $rule1$ and $rule2$ are ground functional terms that name any two rules in the theory. This higher-priority relation given by h_p is required to be irreflexive. The derivability relation, $\vdash$, of the background logic s given by the single inference rule of modus ponens.*

For simplicity, it is assumed 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, i.e. $T \vdash L$ under the background logic. The subset of rules in the argument T that belong to $\mathcal{T}$ is called the *object-level* argument. Note that 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. We call $\mathcal{T}_0$ the background knowledge base.

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 of 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) $(\exists r' \in T_1 \cup P_1, r \in T_2 \cup P_2 \text{ s.t. } T \cup P \vdash h\_p(r, r')) \Rightarrow (\exists r' \in T_1 \cup P_1, r \in T_2 \cup P_2 \text{ s.t. } T' \cup P' \vdash h\_p(r', r))$.

Here $S \vdash_{min} L$ means that $S \vdash L$ and that no proper subset of S implies L. 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') is a counter-argument to another such argument when it derives 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 $L = h\_p(r, r')$ 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 *(respectively* skeptical*) consequence of the theory iff L holds in a (respectively 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.

An agent's argumentation theory will be defined as a theory $(\mathcal{T}, \mathcal{P})$ which is further layered in separating $\mathcal{P}$ into two parts as follows.

Definition 3.4. *An agent's* argumentative policy theory, *T, is a theory $T = (\mathcal{T}, (\mathcal{P}_R, \mathcal{P}_C))$ 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 theory at hand. 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 its object-level decision rules, associated to normal situations (related to a default context) and specific situations (related to specific or exceptional contexts). We call the rules in $\mathcal{P}_R$ (named R in the following) and $\mathcal{P}_C$ (named C), *Default* or *Normal Context Priorities* and *Specific Context Priorities* respectively.

4. Argumentation in Decision Aiding

In a nutshell, an Automated Decision Aiding system implements two mappings that correspond to the steps of the Decision Aiding Process. The first mapping, that corresponds to the problem formulation, is one of the form $\Gamma : Problem \rightarrow < A, V, \Pi >$. The second mapping corresponds to the evaluation model construction and is of the form $M :< A, V, \Pi > \rightarrow < \alpha, D, G, \Omega, R >$.

The first mapping can be implemented by a set of logical rules that associate various parameters, such as the features of the input problem, the situation at hand, the profile of the user etc., with the parameters of the problem formulation. For instance the rule

$$select(A, A_i) \leftarrow feature(P, F_1), \ldots, feature(P, F_n)$$

states that if the problem at hand P, has the features $F_1, \ldots F_n$, then the parameter A of the problem formulation of P is instantiated by the set A_i. In the following we use the notation $C_{A,A_i}(P)$ as a shorthand for the set of conditions that need to be satisfied by problem P in order for the parameter A to be instantiated by the set A_i in the problem formulation of P. Therefore the above rule can be represented more shortly by $select(A, A_i) \leftarrow C_{A,A_i}(P)$.

Having incomplete information about the world, such a model needs to account for the *lack of information*. To cope with this, the Automated Decision Aiding process, has to make *assumptions* about other conditions that may influence the selection of the parameters of the problem formulation. These assumptions and the mode of reasoning associated with them can be captured in the argumentation framework that is used by rules of the following form:

$r_i^A : \ select(A, A_i) \leftarrow C_{A,A_i}(P)$
$r_j^A : \ select(A, A_j) \leftarrow C_{A,A_i}(P)$
$R_{i,j}^A : \ h\_p(r_i^A, r_j^A)$
$R_{j,i}^A : \ h\_p(r_j^A, r_i^A) \leftarrow SC_{A,\{j,i\}}(P)$
$C_{j,i}^A : \ h\_p(R_{j,i}^A, R_{i,j}^A)$

The above set of rules says that, under the conditions C_{A,A_i}, A_i is the *default* parameter selection for A in the problem formulation. If in addition to C_{A,A_i} some special conditions $SC_{A,\{j,i\}}$ hold for the problem at hand, then A_j is selected instead. Similar rules can be written for the other parameters of the problem formulation, ie., for V and Π. Each set of rules that corresponds to each of the parameters A, V and Π of Γ, is denoted by T_A, T_V and T_Π respectively.

The next step in automating the Decision Aiding process is to provide rules creating the mapping between the selected problem formulation and the possible evaluation models. This can also be done along the lines described above. Consider for instance the description of the relation between A in the problem formulation, and the parameter α of an evaluation model. The rules that describe this mapping are of the form:

$r_j^\alpha : \ select(\alpha, \alpha_j) \leftarrow select(A, A_i), C_{\alpha,\alpha_j}(P)$
$r_k^\alpha : \ select(\alpha, \alpha_k) \leftarrow select(A, A_i), C_{\alpha,\alpha_j}(P)$
$R_{j,k}^\alpha : \ h\_p(r_j^\alpha, r_k^\alpha)$
$R_{k,j}^\alpha : \ h\_p(r_k^\alpha, r_j^\alpha) \leftarrow SC_{\alpha,\{k,j\}}(P)$
$C_{k,j}^\alpha : \ h\_p(R_{k,j}^\alpha, R_{j,k}^\alpha)$

Similar rules are added for the other parameters of the evaluation model. The rules for the parameters D, G, Ω, and R that correspond to the rule r_j^α are respectively:

$r_j^D : \ select(D, D_j) \leftarrow select(V, V_i), C_{D,D_j}(P)$
$r_j^G : \ select(G, G_j) \leftarrow select(D, D_i), C_{G,G_j}(P)$
$r_j^\Omega : \ select(\Omega, \Omega_j) \leftarrow select(G, G_i), C_{\Omega,\Omega_j}(P)$
$r_j^R : \ select(R, R_j) \leftarrow select(\Pi, \Pi_i), C_{\Pi,\Pi_j}(P)$

Additional rules (of the type R and C), similar to those that have been described for α, are added to the argumentation theory and enforce different selections for the evaluation model parameters, wherever special conditions hold. The set of rules that are associated with the choice of the parameters of M are denoted by T_α, T_D, T_G, T_Ω, and T_R respectively.

Therefore, the resulting argumentation theory T is the union of the above sub-theories, ie., $T = T_A \cup T_V \cup T_\Pi \cup T_\alpha \cup T_D \cup T_G \cup T_\Omega \cup T_R$. At each cycle of the Decision Aiding process that terminates with a rejection of the recommendations, the reasons for this rejection J are added to T, a new theory $T' = T \cup J$ is constructed, and a new reasoning phase starts, this time with the theory T'. In the following we present an example that illustrates the method.

Example 4.1. *An agent wishes to plan a dinner for this evening. He has four options. He could dine with his girlfriend, with his best friend, alone, or stay at home and order a delivery. The agent prefers dining in a restaurant than staying home, and dining with company than dining alone. In the first two cases, the venue is not as important as the company. When dining alone, the standard of the venue is very important. It must be in fact excellent in order to compensate for the lack of company. For dining very late at night, the agent prefers to dine alone, either out or order his favorite pizza for delivery. However, his decision criterion now becomes the time required for service.*

The set of actions relevant to the evening dinner can be represented as follows:
π_1: `dine_with_girlfriend`
π_2: `dine_with_best_friend`
π_3: `dine_alone(`X`)`, $\quad X \in \{r_1, \cdots r_n\}$
π_4: `order_pizza`

The `dine_alone(`X`)` action stands for a set of actions obtained by instantiating variable X with a specific restaurant.

The dining problem can be captured within the decision aiding model described earlier as follows. The agent can choose between two alternative problem formulations. The first is the formulation $\Gamma_1 = \langle A_1, V_1, \Pi_1 \rangle$ where A_1 is the set of actions π_1, π_2, π_3, V_1 is pleasure and venue standard and Π_1 is a choice (of the best thing to do this evening) or a classification (in excellent or acceptable options) problem statement. A second problem formulation is $\Gamma_2 = \langle A_2, V_2, \Pi_2 \rangle$ where A_2 is the same as A_1, V_2 is the time required for service, and Π_2 is always a choice statement. Note that Γ_2 is applicable only in the case where dining takes place late at night. The construction of the two alternative problem formulations can be described in argumentation as follows, where vs denotes the venue standard. For simplicity, we replace the conditions in the right hand of the rules associated with a specific problem with the problem name.

$r_1^A: select(A, \{\pi_1, \pi_2, \pi_3\}) \leftarrow dinner$
$r_2^A: select(A, \{\pi_3, \pi_4\}) \leftarrow dinner$
$R_{1,2}^A: h\_p(r_1^A, r_2^A)$
$R_{2,1}^A: h\_p(r_2^A, r_1^A) \leftarrow late\_dinner$
$C_{2,1}^A: h\_p(R_{2,1}^A, R_{1,2}^A)$

$r_1^V: select(V, \{pleasure, vs\}) \leftarrow dinner$
$r_2^V: select(V, \{service\_time\}) \leftarrow dinner$
$R_{1,2}^V: h\_p(r_1^V, r_2^V)$
$R_{2,1}^V: h\_p(r_2^V, r_1^V) \leftarrow late\_dinner$
$C_{2,1}^V: h\_p(R_{2,1}^V, R_{1,2}^V)$

$r_1^\Pi: select(\Pi, \{choice, classif\}) \leftarrow dinner$
$r_2^\Pi: select(\Pi, \{choice\}) \leftarrow dinner$
$R_{1,2}^\Pi: h\_p(r_1^\Pi, r_2^\Pi)$
$R_{2,1}^\Pi: h\_p(r_2^\Pi, r_1^\Pi) \leftarrow late\_dinner$
$C_{2,1}^\Pi: h\_p(R_{2,1}^\Pi, R_{1,2}^\Pi)$

After describing how the problem formulation alternatives are generated, the mapping between these formulations and the possible evaluation models needs to be specified. Take for instance the first formulation, Γ_1. The two alternative evaluation models that can be generated from it are M_1 and M_2 as specified below

- $\mathcal{M}_1 : \langle\ \alpha = \{\pi_1, \pi_2\}$,
  $D =$ pleasure,
  $G = \{$pleasure : $\pi_1 >_p \pi_2\}$,
  $\Omega = \emptyset$,
  $\mathcal{R} =$ (choice procedure)$\rangle$

- $\mathcal{M}_2 : \langle\ \alpha = \{\pi_3(X) : X \in \{r_1, \cdots r_n\}\}$,
  $D = \{$venue standard, the associate scale being $Excellent(E)$, $Acceptable(A)\}$
  G=$\{$venue standard: $E >$A$\}$,
  $\Omega = \emptyset$,
  $\mathcal{R} =$ (classification procedure)$\rangle$

The construction of these evaluation models is also modelled in the argumentation theory. The rules that correspond to the construction of M_1 and M_2 are described below, where $nagf$ and $nabf$ stand for "not available girlfriend" and "not available best friend" respectively.

$r_1^\alpha: select(\alpha, \{\pi_1, \pi_2\}) \leftarrow select(A, A_1), dinner$
$r_2^\alpha: select(\alpha, \{\pi_3\}) \leftarrow select(A, A_1), dinner$
$R_{1,2}^\alpha: h\_p(r_1^\alpha, r_2^\alpha)$
$R_{2,1}^\alpha: h\_p(r_2^\alpha, r_1^\alpha) \leftarrow nagf, nabf$
$C_{2,1}^\alpha: h\_p(R_{2,1}^\alpha, R_{1,2}^\alpha)$

$r_1^D: select(D, \{pleasure\}) \leftarrow select(V, V_1), dinner$
$r_2^D: select(D, \{vs\}) \leftarrow select(V, V_1), dinner$
$R_{1,2}^D: h\_p(r_1^D, r_2^D)$
$R_{2,1}^D: h\_p(r_2^D, r_1^D) \leftarrow nagf, nabf$
$C_{2,1}^D: h\_p(R_{2,1}^D, R_{1,2}^D)$

$r_1^G: select(G, \{\pi_1 >_p \pi_2\}) \leftarrow select(D, \{pleasure\}), dinner$
$r_2^G: select(G, \{E > A\}) \leftarrow select(D, \{vs\}), dinner$
$R_{1,2}^G: h\_p(r_1^G, r_2^G)$
$R_{2,1}^G: h\_p(r_2^G, r_1^G) \leftarrow nagf, nabf$
$C_{2,1}^G: h\_p(R_{2,1}^G, R_{1,2}^G)$

$r_1^R: select(R, \{choice\}) \leftarrow select(G, \{choice\}), dinner$
$r_2^R: select(R, \{classif\}) \leftarrow select(G, \{classif\}), dinner$
$R_{1,2}^R: h\_p(r_1^R, r_2^R)$
$R_{2,1}^R: h\_p(r_2^R, r_1^R) \leftarrow nagf, nabf$
$C_{2,1}^R: h\_p(R_{2,1}^R, R_{1,2}^R)$

The following scenario illustrates how the agent could use the above argumentation theory T. When faced with a dining decision, theory T derives Γ_1 as the preferred formulation of the problem. Given Γ_1, the theory generates the evaluation model M_1. The best choice according to M_1 is dining with the girlfriend but the agent discovers that she is not available. The second choice is to dine with the best friend but he is also not available. This means that the condition $nabf$ and $nagf$ are true and therefore the dining decision has to be taken according to the model M_2. By applying M_2 he attempts to find a restaurant considered as excellent but meanwhile he discovers that it is already late evening. This means that $late\_dinner$ special condition is now true. The theory now becomes $T \cup \{nabf, nagf, late\_dinner\}$. The new theory derives that Γ_1 should be abandoned altogether, and selects the problem formulation Γ_2 which leads to the evaluation model

$\mathcal{M}_3 : \langle\ \alpha = \{\pi_3, \pi_4\}$,
$D = \{$ min of estimates of service time for each action $\}$: $min\_time(\pi_3(X) : X \in \{r_1, \cdots r_n\}) = 50min$, $min\_time(\pi_4) = 10min\}$,
$G = \{$time : $x >_t y$ **iff** $t(x) < t(y)\}$,
$\Omega = \emptyset$,
$\mathcal{R} =$ (choice procedure)$\rangle$

and the choice will be action π_4 which finally is implemented. The evaluation model M_3 is generated by rules similar to those for M_1 and M_2.

5. Conclusions

In this paper we presented how the decision aiding process can be automated and demonstrated how it can be embedded in autonomous agents in order to provide decision aiding to a user or replace him in some decision task. Our proposal is a first attempt to provide a formal account of the decision aiding for descriptive but also prescriptive purposes. This formalism is based on a highly expressive argumentation framework we adopted, which allowed us to capture the defeasible character of the cognitive artifacts generation process. A comparison between this specific argumentation framework and other argumentation frameworks can be found in [9].

The specific automated decision aiding model we proposed, gives to the agents the flexibility to choose dynamically evaluation models, and thus the appropriate type of decisions (best choice, ranking or classification of the possible alternatives, etc.) to be taken according to the circumstances at hand.

In this paper the agent is considered as an individual entity, where argumentation is used as a self-deliberation mechanism. However, the same argumentation framework can be used in multi-agent context in order to model agent interaction ([7]). In this case, the revision of the produced artifacts of the decision aiding process can be triggered by the defeat, during an inter-agent argumentation-based interaction, of the arguments involved in the construction of these artifacts.

Future research will investigate further the relation between argumentation and decision aiding in order to identify possible uses of decision aiding concepts for knowledge representation purposes. Additionally, further work is needed to develop a systematic methodology for building agents based on the argumentation theories used in this paper.

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] P. McBurney and S. Parsons, 'Argumentation-based communication between agents', in *Communication in Multiagent Systems*, volume 2650 of LNAI, Springer-Verlag, 2004.

[3] V. Belton and T. Stewart, *Muliple Criteria Decision Analysis: An Integrated Approach*, Kluwer Academic, Dordrecht, 2002.

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

[5] S. French, *Decision theory - An introduction to the mathematics of rationality*, Ellis Horwood, Chichester, 1988.

[6] A.C. Kakas, P. Mancarella and P.M. Dung, 'The acceptability semantics for logic programs', in *Proceedings of ICLP94*. 1994.

[7] A.C. Kakas, N. Maudet and P. Moraitis, 'Layered Strategies and Protocols for Argumentation Based Agent Interaction', in *Proc. AAMAS'04 1st International Workshop on Argumentation in Multi-Agent Systems, (ArgMAS'04)*, Springer LNAI, (to appear).

[8] A.C. Kakas and P. Moraitis, 'Argumentation based decision making for autonomous agents', in *Proceedings of AAMAS03*, pp. 883–890. ACM Press, New York, (2003).

[9] A.C. Kakas and P. Moraitis, 'Argumentative Agent Deliberation, Roles and Context', in *Electronic Notes in Theoretical Computer Science (ENTCS)*, **70**(5), 2002, Elsevier Science Publishers.

[10] R.L. Keeney, *Value-Focused Thinking. A Path to Creative Decision Making*, Harvard University Press, Cambridge, 1992.

[11] S. Parsons, C. Sierra, and N.R. Jennings, 'Agents that reason and negotiate by arguing', *Journal of Logic and Computation*, **8**, 261–292, (1998).

[12] H. Prakken and G. Vreeswijk, 'Logics for defeasible argumentation', in *Handbook of Philosophical Logic, Vol 4*, eds., D. Gabbay and F. Guenthner, 219–318, Kluwer Academic, Dordrecht, (2002).

[13] J. Rosenhead, *Rational analysis of a problematic world*, J. Wiley, New York, 1989. 2nd revised edition in 2001.

[14] A. Tsoukiàs, 'Preference modelling as a reasoning process: a new way to face uncertainty in multiple criteria decision support systems', *European Journal of Operational Research*, **55**, 309–318, (1991).

[15] 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.

[16] A. Tsoukiàs, 'On the concept of decision aiding process', Technical Report 2003-38, DIMACS, Rutgers University, (2003).

[17] D. von Winterfeldt and W. Edwards, *Decision Analysis and Behavorial Research*, Cambridge University Press, Cambridge, 1986.

[18] Gorgias: A system for argumentation and abduction, 2002. `http://www.cs.ucy.ac.cy/ nkd/gorgias`