ABA: Argumentation Based Agents*

A. Kakas[1], L. Amgoud[2], G.Kern-Isberner[3], N. Maudet[4], and P. Moraitis[5]

[1] University of Cyprus, antonis@ucy.ac.cy
[2] IRIT-CNRS at Toulouse, amgoud@irit.fr
[3] University of Dortmund, gabriele.kern-isberner@cs.uni-dortmund.de
[4] University Paris 9 Dauphine, maudet@lamsade.dauphine.fr
[5] Paris Descartes University, pavlos@mi.parisdescartes.fr

Abstract. Many works have identified the potential benefits of using argumentation in multiagent settings, as a way to implement the capabilities of agents (eg. decision making, communication, negotiation) when confronted with specific multiagent problems. In this paper we take this idea one step further and develop the concept of a fully integrated argumentation-based agent architecture. Under this architecture, an agent is composed of a collection of modules each of which is responsible for a basic capability or reasoning task of the agent. A local argumentation theory in the module gives preferred decision choices for the module's task in a way that is sensitive to the way the agent is currently situated in its external environment. The inter-module coordination or intra-agent control also relies on a local argumentation theory in each module that defines an internal communication policy between the modules. The paper lays the foundations of this approach, presents an abstract agent architecture and gives the general underlying argumentation machinery minimally required for building such agents, including the important aspects of inter-module coordination via argumentation. It presents the basic properties that we can expect from these agents and illustrates the possibility of this type of agent design with its advantages of high-level of flexibility and expressiveness.

1 Introduction

In recent years, many authors have promoted argumentation as a means to deal with specific multi-agent problems, for instance negotiation or communication with other agents. Indeed, recently argumentation has seen its scope greatly extended, so that it now covers many of the features usually associated to the theories of agency [28]. The benefits of argumentation are well established: a high-level of flexibility and expressiveness, allowing powerful and diverse reasoning tasks to be performed. In particular, different semantics can be used for different purposes without altering the underlying basic principles.

* This work grew out of the initiative of the 2008 Dagstuhl Workshop on the "Theory and Practice of Argumentation Systems" to ask groups of researchers to propose ways of consolidating the work on several main themes of argumentation in Computer Science, such as the theme of argumentation in agents, which is the concern of this paper.

We study how this idea can be taken one step further to develop the concept of a fully integrated argumentation-based agent (ABA) architecture. The idea seems natural: for instance, to make the most of argumentation-based protocols, an agent should also demonstrate some argumentative reasoning capabilities. Similarly, for an agent to take informed and coherent decisions it needs to be able to argue about its choices by linking them to its underlying motivations and needs. What is missing is a global framework of how all these features could be glued together, both in terms of abstract design and technical specifications.

This paper lays the foundations of such an approach to agency, presents an abstract agent architecture obeying these principles, and gives the general underlying argumentation machinery minimally required for building such agents. In short, an agent is made of a collection of modules each of which is responsible for a basic capability or reasoning task of the agent. This is governed by a local argumentation theory in the module that gives preferred decision choices for the local task of the module, sensitive to the way the agent is currently situated in its external environment. The inter-module coordination and thus intra-agent control also relies on an argumentation theory that defines an internal communication policy between the modules. This gives an agent architecture that is coherently designed on an underlying argumentation based foundation.

From the early BDI architectures to the recent developments of computational logic based agents, the genealogy of agent architecture is now very dense. We can summarize the main objectives sought by the latest developments of agent architectures as follows:

- make the design easier (for instance by adopting readily understandable languages, or by semi-designing the agent, like introducing typical agent types [19]);
- bridge the gap between specification and implementation, the most typical case being the first BDI specifications vs. its concrete implementations (as noticed for instance by [22]);
- make agents more flexible and sensitive to external events [24], in particular going further than the classical "observe-think-act" cycle (as for instance the cycle theories in the KGP model [12] do);
- introduce new features not originally present in the architectures that now appear to be vital to autonomous agents (for instance social features [6] or learning [1])

We regard the adoption of a unified argumentation based architecture as highly positive regarding the first three issues, in particular. Our argumentation-based agent architecture is a high-level architecture that can also encompass other methods by transparently incorporating them in the architecture as black boxes that generate information or choices to be argued about. Its main concern is indeed to manage its different options by considering the arguments for and against in the light of the currently available information from the environment.

The argumentation basis of the ABA architecture does not depend on any specific argumentation framework but only requires some quite general properties of any such framework to be used. Irrespective of the framework used the

argumentation-based foundation of ABA agents provides various advantages, including that of its rational or valued based decisions that facilitates the focus of purpose by the agent and the more effective interaction between agents which can explain their positions or requests.

Our work shares similarities with other argumentation based agent approaches when it comes to addressing specific issues and features of agents, e.g. in the recent KGP model of agency [12] goal decision and cycle theories for internal control are also captured through argumentation. However, the objective of assembling all these features in a single and coherent architecture uniformly based on argumentation has been the main challenge of our approach. The closest connection is with the work in [27] which proposes an Agent Argumentation Architecture (called AAA) and further developments of this in [16]. As in our case, argumentation is used as the primary means to arbitrate between conflicting motivations and goals. More specifically, in this work the high-level motivations of the agents are operationally controlled by *faculties*. These faculties make use of a dialogue game to arbitrate among the conflicting goals, depending on the consequences they foresee, or on favoured criteria of assessment. Also the recent work of Argumentative Agents [25] with their ARGUGRID platform uses argumentation as the main way to support an agent's decisions with particular emphasis on the process of negotiation between such agents.

The wider context of our work is that of modularly composed agent architectures with internal rationality for managing the different internal processes of the agent, as found for example in the works of [23] and [20]. In our proposed approach argumentation plays a central role both for the decisions within each module and for the interaction between the various agent modules. In particular, our approach offers an alternative way to view and possibly extend the use of bridge rules that other architectures use for the intra-agent reasoning.

The rest of the paper is as follows. In the next section we present the basic argumentation machinery for building ABA agents. Sections 3 and 4 present the abstract agent architecture and its intra-agent control. In Section 5, we detail some basic formal properties that we can expect from ABA agents, concluding in Section 6.

2 Argumentation Basics

The backbone of an ABA agent is its use of argumentation for decision making. Argumentation allows an agent to select the "best" or sufficiently "good" *option*(s), given some available information about the current state of the world and the relative benefits of the potential options. For instance, an agent may want to decide its best options of goals to pursue or partners to work with. We will denote with $\mathcal{O}$ the set of possible options of a decision problem. For simplicity of presentation, these options are assumed to be mutually exclusive and pairwise conflicting. For instance, an agent may want to choose between two possible partners, Alice and Carla, for carrying out a task. Thus, $\mathcal{O} = \{\text{Alice}, \text{Carla}\}$.

The overall value of any certain option can be judged through evaluating by means of several *parameters* how much this option conforms to the preferences of the decision maker. An agent may for instance choose between Alice and Carla on the basis of parameters such as reliability and generosity. Each agent is thus equipped with finite sets, $\mathcal{M}$, of parameters that are used in expressing the relative preferences or priority amongst options. This, as we will see below, is done using these parameters to parameterize the various options and the arguments that the agent has for these (c.f. with the Value-Based Argumentation in [2]). Parameters may not be equally important, for example the reliability of a partner may be more important than its generosity. Thus arguments for a partner that carry the parametrization of reliability will be preferred. We will denote by, $\geq$, a partial ordering relation on a set $\mathcal{M}$ of parameters reflecting their importance.

From the current state of the world, as perceived by an agent, *basic arguments* are built in favor of options in $\mathcal{O}$ and these are labelled using appropriate parameter spaces, $\mathcal{M}$, of the agent. Let $\mathcal{A}$ denote the set of all those arguments for a specific decision problem. Each argument supports only one option but an option may be supported by many arguments. Let $\mathcal{F} : \mathcal{A} \longmapsto \mathcal{O}$ be a function which associates to each argument, the option it supports. An argument highlights the positive features of each option, such as the parameters that label the option. For example, an argument in favor of Carla would be that she is generous, while an argument in favor of Alice would be that she is reliable. Let also $\mathcal{H} : \mathcal{A} \longmapsto (2^{\mathcal{M}})$ be a function that returns the parameters that label each argument. Since the parameters are not necessarily equally important, the arguments using them will in general have different strengths. For instance, if we assume that reliability is more important than generosity, then the argument that is based on reliability is stronger than the one that is based on generosity.

We will assume that the relative strength between arguments is based on the an underlying priority ordering on the parameter space that is used to label the arguments. Hence in what follows, $\succeq$ will denote a partial preorder on the set of arguments that expresses the relative strength of arguments, grounded in some way on the relation $\geq$ on the parameter space of arguments. This lifting of the ordering on the parameters to an ordering on the arguments, that are labelled by the parameters, can be done is several ways and is in general application domain depended.

In most frameworks for argumentation we have two basic components: a set, $\mathcal{A}$, of arguments and an *attack* relation among them. This relation captures the notion of one argument conflicting with another and providing a counter-argument to it. In our case, arguments that support distinct options are conflicting since the options are assumed to be mutually exclusive. So, e.g., we might define that α_1 `Attacks` α_2 iff $\mathcal{F}(\alpha_1) \neq \mathcal{F}(\alpha_2)$, and $\alpha_1 \succeq \alpha_2$, for two arguments $\alpha_1, \alpha_2 \in \mathcal{A}$. This gives the following argumentation theory:

Definition 1 (ABA Argumentation theory). *An argumentation theory,* `AT`*, for decision making of an ABA agent is a tuple* $\langle \mathcal{O}, \mathcal{A}, \mathcal{M}, \mathcal{F}, \mathcal{H}, \geq, \succeq,$ `Attacks`$\rangle$

where `Attacks` *is chosen by the specific argumentation framework that we base the agents on.*

The process of argumentation is concerned with selecting amongst the (conflicting) arguments the *acceptable* subsets of arguments. This notion of acceptability has extensively been studied by several papers, e.g. [8]. Indeed, there are different proposed semantics for evaluating arguments and the semantics of (maximal) acceptable arguments. One widely used form of such a semantics is based on the notion of *admissible* arguments. According to this semantics, a subset $\mathcal{B}$ of $\mathcal{A}$ is admissible and hence acceptable iff it satisfies the following requirements:

- it is not self attacking, i.e. there is no element of $\mathcal{B}$ that attacks another element of $\mathcal{B}$,
- for every argument $\alpha \in \mathcal{A}$, if α attacks (w.r.t. `Attacks`) an argument in $\mathcal{B}$, then there exists an argument in $\mathcal{B}$ that attacks an argument in $\mathcal{A}$.

Maximal admissible arguments, called *preferred extensions*, are then taken as the maximal acceptable extensions of a given argumentation theory. In an argumentation-based approach, the choice of the "best" option(s) among elements of $\mathcal{O}$ is based on the maximal acceptable arguments associated with the different options as follows.

Definition 2 (Best decision/option(s)). *Let* $\texttt{AT} = \langle \mathcal{O}, \mathcal{A}, \mathcal{M}, \mathcal{F}, \mathcal{H}, \geq, \succeq, \texttt{Attacks} \rangle$ *be an argumentation theory for decision making,* $\mathcal{E}_1, \ldots, \mathcal{E}_n$ *its maximal acceptable extensions, and* $d \in \mathcal{O}$*. The option* d *is a possible best (or optimal) decision of AT iff* $\exists \alpha \in \mathcal{A}$ *such that* $\mathcal{F}(\alpha) = d$ *and* $\alpha \in \mathcal{E}_i$ *for some* $i = 1, \ldots, n$*.*

It is clear that the basic component of this decision theory is the preference relation $\geq$ on the set $\mathcal{M}$ of parameters. This relation may be context dependent on the current situation in which the deciding agent finds itself. For example, the preference of reliability over generosity applies in case the task to do is urgent, while generosity may take precedence over reliability in case the agent is short on resources (money). Furthermore, conflicts between preferences may arise, e.g when an agent is in a situation in which it has an urgent task and it lacks resources. Then our original decision problem for choosing an optimal option is elevated to the decision of which of the preferences is (currently) more important. We are thus faced with a new decision problem on choosing the best priority amongst the basic arguments to answer our original decision problem.

This new problem is of the same form as the decision problems that we have described above where now our options have the special form $m \geq m'$ or its conflicting one of $m' \geq m$, where m and m' are members of $\mathcal{M}$, or of the form $\alpha \succeq \beta$ where α and β are arguments, i.e. members of $\mathcal{A}$. Our argumentation theory thus contains *priority arguments* for these options capturing *higher order preferences*. We can then combine these two argumentation theories to have a single argumentation theory that contains both basic arguments for the object-level options and priority arguments for the relative importance of the parameters and arguments. This extension can be done in several ways, see e.g.

[15, 21, 14, 7]. In [21], where this problem was originally studied, basic (object-level) arguments are constructed from rules which are given names or classified in types and then preference arguments are given as rules for a priority ordering between (the names of or the types of) two rules. Such priority rules can also be named or categorized and hence high-order preference can be given as rules for the priority between (lower-level) priority rules.

3 ABA Architecture

The ABA architecture's basic principle is to build an agent from a loosely coupled set of modules that are to a large extent independent from each other with no or minimal central control. Each module is based on an argumentation theory, concerning a certain internal task of the agent, that provides a policy of how to take decisions on this type of tasks. A module contains also another argumentation theory responsible for its involvement in the intra-agent control (IAC) of the agent. Together these local IAC theories in each module give (see the next section) a distributed high-level argumentation-based communication protocol under which the internal operation of the agent is effected. The modularity of the ABA agent approach aims to allow the easy development of an agent by being able to develop separately its modules adding further expertise to it as we see appropriate without the need to reconsider other parts of the agent. An ABA agent module is defined as follows.

Definition 3 (ABA Agent Module). *An ABA agent module is a tuple* $M = \langle IAC, T, R \rangle$ *where:*

- IAC *is an argumentation theory for intra-agent control,*
- T *is an argumentation theory for the task of the module,*
- $R = \langle P, C \rangle$ *where* P *and* C *are sets of names of other modules, the parent and child modules of* M *respectively.*

Each module, M, is based on its own argumentation theory, T, pertaining to its specialized task. This is an expert (preference) policy comprising, as we have described in the previous section, of arguments for the different choices parameterized in terms of preference criteria together with priority arguments on the relative importance of these criteria and hence also on the basic arguments that they parametrize. The information (basic and priority arguments) contained in the argumentation theories in the various modules is given to the agent at its initial stage of development and remains relatively static, although some parts may be further developed during the operation of the agent. The dynamic information of the agent is that of its view of the external world, as we shall see below. This also affects which part of the static information is applicable in each situation.

The sets P and C of a module express a dependence between the modules that captures a request-server relationship where the decisions taken by a parent module form part of the problem task of a child module. For example, a Planning module will be a child of a Goal Decision module since Planning

decides on (or selects plans) to achieve the goals decided by Goal Decision. The IAC component will be described in more detail in the next section.

Definition 4 (ABA Agent). *An ABA agent is a tuple,* $\langle Ms, Mot, WV \rangle$*, where*

- $Ms = \{M_1, ..., M_n\}$ *is a set of ABA modules for the different internal capabilities of the agent,*
- Mot *is a module containing an ABA argumentation theory for the agent's Motivations and Needs,*
- WV *is a module that captures the current World View that the agent has about its external environment.*

The number of modules and the capability they each provide to the agent is not fixed but can vary according to the type of application that the agent is built for. However, the Motivations and Needs (Mot) and the World View (WV) modules are specialized modules that play a central role and are arguably required to design any ABA agent.

Motivations and needs. An ABA agent contains a special module, Mot, for governing its high-level Motivations and Needs. These in turn can play a role in the decisions of many different modules of the agent. The Mot module comprises of an ABA argumentation theory where, through a preference structure on the Needs of the agent that are parameterized by its Motivations and that also depends on the current world view of the agent, it decides on the current high-level Needs of the agent. It thus defines the current *Desires* of the agent that drive the behaviour of the agent. This is achieved through the use of Needs as a parameter space for the arguments in many of the other modules. For example, the concrete goals that an agent sets in its Goal Decision module are selected according to these desires and therefore they come to best serve these desires. One way to formulate the Motivations and Needs policy is to follow a cognitive psychology approach. In particular, as in [14], we can use Maslow's basic motivations $M_1, \ldots, M_5$ for human behaviour: M_1 = *Physiological*, M_2 = *Safety*, M_3 = *Affiliation or Social*, M_4 = *Achievement or Ego*, and M_5 = *Self-actualization or Learning*. The motivations policy is then an argumentation theory for the relative priority or strength of these motivational factors, dependent on the current world view.

Example 1. Consider Alice and her friends $\mathcal{A} = \{\text{Bill, Carla, Dave, Elaine}\}$. Let us suppose that Alice's current needs are $\mathcal{N}_{\mathcal{A}} = \{need_f, need_c, need_m, need_e\}$, where $f = food, c = company, m = money, e = entertainment$. The arguments for these may be labelled by the basic motivations in the following way: $\mathcal{H}(need_f) = \{M_1\}$, $\mathcal{H}(need_c) = \{M_3\}$, $\mathcal{H}(need_m) = \{M_2\}$, $\mathcal{H}(need_e) = \{M_5\}$. We will assume that the induced strength relation on the basic arguments for Alice's current needs renders the arguments for the needs of $food$, *company* and *money* acceptable, while the argument for *entertainment* is not. These acceptable needs form the current *desires* of Alice and are part of her current state. These then affect the argumentation in other modules of Alice which use the Needs to parameterize their arguments.

Example 2. Alice decides the high-level goals to serve these desires in her GOAL DECISION module. Given her current World View, she has basic arguments for the following set $\mathcal{D}_A$ of potential goals:

$$\mathcal{D}_A = \begin{cases} G_{cheap} : \text{Have a } \textit{cheap} \text{ dinner with company} \\ G_{free} : \text{Be taken out for dinner by someone} \\ G_{home} : \text{Have dinner alone at } \textit{home} \end{cases}$$

From the connections between goals and needs the basic arguments for these potential goals are labelled by the needs they each serve:

$$\begin{array}{lll} A_c \text{ with } & \mathcal{F}(A_c) = G_{cheap} & \text{and } \mathcal{H}(A_c) = \{need_f, need_c\} \\ A_f \text{ with } & \mathcal{F}(A_f) = G_{free} & \text{and } \mathcal{H}(A_f) = \{need_f, need_c, need_m\} \\ A_h \text{ with } & \mathcal{F}(A_h) = G_{home} & \text{and } \mathcal{H}(A_c) = \{need_f\} \end{array}$$

Alice makes use of her argumentation theory for determining the priority of these arguments by evaluating the parameter pertaining to each argument. This yields $A_f \succeq A_c \succeq A_h$, and so G_{free} is the only goal that has an acceptable argument and this is the current choice in the GOAL DECISION module.

Example 3. In order to achieve her goal G_{free}, Alice adopts a preferred plan Π_{free} —choice of restaurant, time of dinner etc. — from her plan library in a similar argumentation process. She chooses this plan using her argumentation theory for plan selection in her Plan module based on some parametrization of the plans and a priority ordering of these parameters. The chosen plan cannot be effected entirely by Alice as it requires resources from other agents (it contains the requests for the external resources for *money*, (req_m), and for *company*, (req_c)). Now Alice is faced with the problem of deciding which other (sets of) agent can best serve these requests. This is the task of the COLLABORATION module. In this she has arguments for different agent partners to provide needed resources. These arguments are labelled by a parametric space of agent profiles, such as: $\mathcal{M}_{profile}$ = {Reliable, Likeable, Generous, Boring, Parsimonious, Offensive, Wealthy} = $\{R, L, G, B, P, O, W\}$. In Alice's world view each of the other agents have a profile parametrization, e.g.: $\mathcal{P}_A$(Bill) = $\{R, P, B, W\}$, $\mathcal{P}_A$(Carla) = $\{R, L\}$, $\mathcal{P}_A$(Dave) = $\{O, G, B, W\}$. Alice's argumentation policy for the priority of arguments for the different partner agents makes use of these profile parameters by measuring the extent to which the profiles serve the requested resources. Here Dave is the only agent that has profile attributes (G, W) that serves req_m, and so there is just one acceptable argument and corresponding choice of Dave.

World view. The agent's world view is maintained in the WORLD VIEW module, WV, providing a common view of the current state of the world to all other modules of the agent. The basic arguments and priority arguments in the agent's other modules depend on the world view, thus making them context dependent and adaptable to changes in the external environment of the agent. The WORLD VIEW module is thus a special module in the ABA architecture responsible for this global task. It can be realized in different ways, e.g. in terms

of beliefs and a process of belief revision as in a BDI architecture. Then the current beliefs give the current world view that grounds the arguments in the different modules of the agent. Nevertheless, the WV module can can also be based, if the designer so wishes, on an argumentation theory for REASONING ABOUT ACTIONS AND CHANGE, as shown for example in [13, 26]. In this the main arguments are those of forward and backward persistence in time of world properties and the preference structure is given by the time ordering of the times from which the persistence starts, e.g. forward persistence that is rooted at later time is stronger than other forward persistence rooted at an earlier time and analogously for backward persistence. The external environment feeds this module with new information on events and properties that have been observed at certain times. An argumentation process then gives the properties of the world that currently hold.

Figure 1 gives a picture of the overall general structure of the basic architecture of an ABA agent. During its operation an ABA agent is characterized by a current *internal state*.

Definition 5 (Agent State). *A state of an ABA agent, $\langle Ms, Mot, WV\rangle$, is a tuple $\langle V, \mathcal{D}\rangle$ where:*

- *V represents the current view of the world as given by WV,*
- *$\mathcal{D} = \{CS_{M_1}, ..., CS_{M_n}\}$ where each, CS_{M_i}, is a tuple $\langle D, L, S\rangle$, representing the current state of the module M_i, where D is its current decision, as given by its argumentation theory, T_i, L is the level of commitment on D and $S \in \{keep, abandon\}$ is the current status of the decision D.*

The level of commitment and status of a module's decision are maintained by the intra-agent control, IAC theory of the module, as we see in the next section.

Feasibility arguments. In deciding the status of a decision it is useful to make a distinction between *feasibility* arguments and *optimality* arguments that an agent can have against a given decision. Feasibility arguments attack the feasibility of a given decision based on current world view information (*e.g.*, the agent may learn that the server it tries to connect to is down), while *optimality* arguments are situation independent arguments for the value of a given decision (*e.g.*, the agent may prefer servers whose storage capacity of the server is above a certain threshold). Part of the world-view module will then enable feasibility arguments specific to the "reality of the situation" for the current decision. Typically, feasibility arguments will parameterize decisions as being: *available*, *currently unavailable* (the current world-view discards this decision but it may be available again later on), or *unavailable* (the world-view discards this decision for ever). These new arguments, $\mathcal{A}^{fea}(V)$, for (or mostly) against the current decision, enabled in the new world view V of the agent, will affect the (meta-level) decision of the IAC theory to keep or abandon the module's decision.

4 Intra-agent Control

The intra-agent control (IAC) of an ABA agent is effected through a *communication protocol* that governs the interaction between the different modules of the

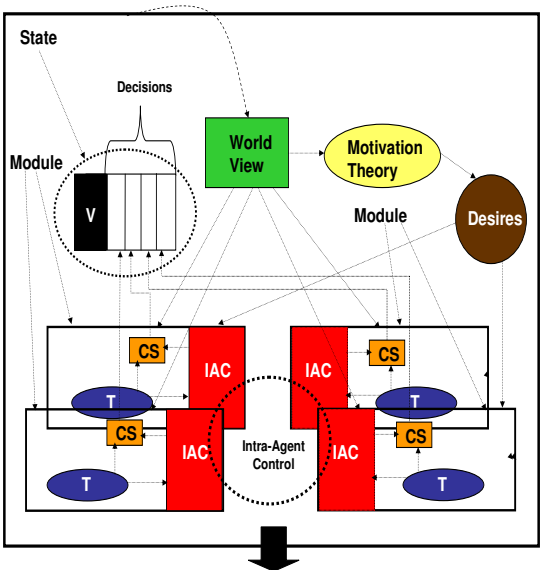

Fig. 1. ABA Architecture

agent. Through this protocol the modules pass messages between them (from parent to child and vice-versa) that in effect determine a distributed flow of control of the agent. For example, the Goal Decision module when it has decided on a new preferred goal it would send a message to its child module of Plan Decision, so that it would start the process of finding a preferred plan for it. Similarly, when a current (preferred) plan becomes untenable then the Plan Decision module would either decide on a new plan or inform the Goal Decision module thus prompting it to reevaluate and perhaps abandon this goal. As such there is no central control per se, except a mechanism for noting in the world view of the agent the passage of time and the changes in its external environment and distributing this to the other modules.

The IAC communication protocol is realized by endowing each of its modules its own ABA argumentation theory, IAC, responsible for governing its communication with the other modules. The basis of each of these IAC theories is (i) to decide *when to reconsider*, in the light of new information coming from the external environment either directly by a change in the current world view or indirectly through messages from other modules, the current decision of the module; and (ii) to decide *how to reconsider* these decisions, examining whether to *abandon or keep* them. Hence, the IAC as a whole, is responsible for updating the set $\mathcal{D}$ of current decisions in the internal state $\langle V, \mathcal{D} \rangle$ of the agent as its world view, V, changes. The IAC theories are argumentation theories of the following form.

Definition 6 (IAC Argumentation Theory). *The intra agent control theory of a module, M, is a tuple $\langle T_L, P_{Status} \rangle$ where:*

- *T_L is a theory for defining the commitment level, L, for the (object-level) decisions in M,*
- *P_{Status} is an ABA argumentation theory for the options $Keep(D)$ or $Abandon(D)$, with D a decision in M, that uses the commitments levels of T_L as parameters of its arguments.*

The levels of commitment, given by T_L, form (part of) the parametric space for the intra-agent control argumentation theory, P_{Status}, of the module. The arguments in P_{Status} for keeping or not a decision can be annotated (or even expressed) in terms of relative changes in these levels of commitment as time passes and new information from the external environment is acquired. The specific parameter space for the commitment levels and the type of theory T_L that assigns these are open to the designer. Nevertheless, the argumentation basis of an ABA agent under which its decisions are taken by its modules in the first place, allows us to define a natural form of commitment as follows.

Definition 7. *Let D be a decision of a module and $T(V)$ denote the module's argumentation theory T grounded on the current world view V. Then the current* commitment level *for D is given as follows:*

- *Level 5, iff D is uniquely sceptically preferred by $T(V)$, i.e. D holds in all maximal acceptable extensions of $T(V)$*
- *Level 4, iff D is credulously preferred by $T(V)$, i.e. D holds in one but not all maximal acceptable extension of $T(V)$*
- *Level 3, iff D does not hold in any acceptable extension of $T(V)$ but there exists a basic argument for D*
- *Level 2, iff D does not have a basic argument in $T(V)$*
- *Level 1, iff neither D nor any other alternative decision D' hold in any maximal acceptable extension of $T(V)$*

Hence the commitment level is a measure of the degree of acceptance (or optimality) of the decision with respect to the agent's optimality arguments for and against this decision in the argumentation theory T of the module. As the world view of the agent changes the structure of the module's argumentation theory, T, changes since different arguments and a different subset of the parameters that annotate the arguments are applicable. This then changes the degree of acceptance of the decision and hence its commitment level.

When and how to reconsider? The reconsideration of the commitment level of the current decision in a module every time that we apply the P_{Status} theory can be computationally non-effective. Under the above definition of commitment, the argumentation reasoning needed to reexamine the degree of acceptance of a decision can in general be costly. Hence to make the operation of P_{Status} more practical we can layer its decision process into two stages. In the first stage we apply a lightweight *Decision Reconsideration* policy that efficiently tells us whether we indeed need to reconsider the current decision. Only if the result from this is affirmative we continue to consider the full P_{Status} reasoning for deciding the fate of the current decision. Otherwise, we keep the current decision. The

Decision Reconsideration policy can be effectively constructed by considering a set of testing conditions that can trigger the possibility for a change in the level of commitment or degree of acceptance when this forms the commitment level. To be more specific, the degree of acceptance of a decision, D, in a module might decrease if new optimality arguments either against D, or in favour of another decision D' are enabled by V. Reconsideration should also be sensitive to the fact that a new feasibility argument against D, in $\mathcal{A}^{fea}(V)$, generated by a new world view, V, occurs. Likewise, the disabling in V of an argument in favour of D may lead to a reconsideration, and similar conditions for priority arguments can be specified. The *cautiousness level* specifies to which of these inputs the agent triggers the reconsideration process. Other factors may be used in this policy, in particular the *time* elapsed, denoted by t, from the time, t_0, that a decision was taken initially, with two important thresholds: t_α before which we have enough time to replace the decision and t_β after which it is too late to replace the decision ($t_0 < t_\alpha < t_\beta$). This allows us to design ABA agents with different characteristics whose operational behaviour can vary across the whole spectrum of "open" to "blind" BDI like agents and whose operation can be dynamically adapted to external changes. An "open" agent would be given by setting $t_\alpha = t_\beta = \infty$ whereas a "blind" agent by setting $t_\alpha = t_\beta = t_0$.

The role then of the argumentation theory component, P_{Status}, of the IAC theory, is to decide whether to keep or abandon the current (task) decision of the module by reexamining its commitment level or in effect by reexamining its degree of acceptance in the face of new information. The basic arguments of P_{Status} (denoted by $Arg([Keep|Abandon], D, level_1, level_2)$ can be built using the following underlying form:

- $keep(D)$ **if** the level of commitment of D is the same or increases
- $abandon(D)$ **if** its level of commitment decreases.

Example 4. The following arguments may define the default behaviour of a module of Alice: [Arg(Keep,D,5,4)] for keeping a decision D when its commitment level has fallen from 5 to 4 (since the decision is still acceptable in the module's theory) or an argument [Arg(Abandon,D,any,3)] for abandoning a decision when its commitment level falls to level 3 (as the decision is now not acceptable). Note though that there can be special circumstances, e.g. special types of decisions or extreme cases of the world view, when the opposite arguments might apply.

The argumentation reasoning of P_{Status} also depends on the current relevant feasibility arguments. For example, a child module may inform its parent module that the child's current decision is now at commitment level 1, i.e. that it can find no solution to the current problem that the parent module has sent it. This may be the result of information that the child module has received from the environment and/or from other modules. Thus a new feasibility argument is enabled in the parent module's P_{Status} theory, denoted by $[Arg(Abandon, D, c - unavailable)]$, for giving up its current decision D, for which it is informed that *currently* it cannot be effected in any way. The newly enabled feasibility arguments in P_{Status} can then be compared, via priority arguments in P_{Status}, with the other arguments based on the commitment level

reexamination considered above. For example, should a module abandon its decision when it is informed by a child module that this cannot be (currently) achieved, even if its commitment level for this decision remains at the highest level? In other words, which is the stronger argument amongst the two basic arguments of [Arg(Keep,D,5,5)], which is based on the subjective evaluation of D, and [Arg(Abandon,D,c-unavailable)] based on objective information and under what conditions this is so? The preference structure of P_{Status} addresses such questions so that the IAC can weight up such different factors.

Example 5. We may capture the (default) preference to abandon currently unattainable decisions but not so when they are still optimally the most preferred ones with the priority arguments: [Pr1-Arg(Abandon,Keep)]: [Arg(Abandon,D, c-unavailable)] $\succ$ [Arg(Keep,D,L1,L2)] **if** $L2 \neq 5$ and [Pr2-Arg(Keep,Abandon)]: [Arg(Keep,D,L1,5)] $\succ$ [Arg(Abandon,D, c-unavailable)]. Of course, we may want to condition the second priority on the condition that there is still enough time for the world to change and make the decision D available again, e.g. for a collaborating agent to change its mind and make itself available.

With such priority arguments and the preference structure that follows from them, the designer of an ABA agent can give it a general strategy of operation, a characteristic of how to behave when the agent realizes that the implementation of its decisions in the external world has difficulties. Various factors relating to the cost or feasibility of replacing a decision can also be taken into account. For instance, the default argument to abandon decisions when they become relatively sub-optimal can be counter-balanced using another default argument for keeping decisions (as we want to also minimize loss of effort already done), such as: $[Arg(keep, D, default)]$: $keep(D)$ **if** $expensive(D)$, where $expensive(D)$ is application dependent designating which (types of) decisions are costly to discard.

Example 6. To illustrate the various features of the IAC consider again the Alice example and suppose that Alice finds out that Dave has lost all his money and so W will not be in Dave's profile anymore. This disabling of an argument in favour of *Dave* can trigger the reconsideration, in the IAC theory of her COLLABORATION module, of her current decision for Dave. The decision to abandon or keep this decision depends on whether there are still acceptable arguments, w.r.t. the module's (task) argumentation theory, for Dave assigning commitment level at least 4 now, or whether there is no acceptable argument for Dave any more assigning commitment level 3 to him. Other feasibility arguments, e.g. arguments related to the time left before dinner, can also play a role in this decision. Should Alice decide to abandon Dave and the COLLABORATION module has no other choice of partner with an acceptable argument, then the parent module, i.e. the PLAN module, will be notified which in turn will reconsider its current choice of plan using its own IAC theory. Similarly, this may eventually lead to GOAL DECISION module, to re-evaluate its current choice of goal and perhaps abandon this for a new goal to have a cheap dinner, or eat at home.

In general, the reconsideration of decisions and how this is communicated amongst the different parent and children modules of the agent will give an emergent behaviour on the operation of the agent. Under an ideally suited environment we expect that the IAC theory will induce a given pattern of operation on the agent, as we find in many of the proposed agent architectures, e.g. the fixed "Observe-Think-Act" cycle or the more general dynamic cycles given by the cycle theories of the KGP agents defined in [12]. In non-ideal conditions the particular operational behaviour of the ABA agent will be strongly dependent on these IAC theories in its modules.

The communication between modules based on the reconsideration of their decisions and subsequent messages that they send and receive between them can be defined as a form of an internal dialogue policy between the modules. In general, these control dialogue policies can be relatively simple. Nevertheless, it is important that the dialogues generated conform to several required properties of the operation of the agent, e.g. that there is no deadlock (where one module is waiting for a response from another module). We can then draw from the large literature on agent dialogue to ensure such consistency properties of the internal module dialogues. In particular, many of these approaches, e.g. [18, 3] are themselves based on argumentation and hence the link can be made more natural.

5 Properties of ABA Agents

ABA agents are designed so that their operation is based on informed decisions. The working hypothesis that underlies their operation is that the argumentation policies in an agent's different modules capture optimal solutions of the respective decision problems. The argumentation reasoning that they apply in taking their various decisions is such that agents evaluate the current alternatives against each other by comparing the reasons for and against these alternative choices. The acceptable choices in any module are meant to capture the best solutions available at the time. Hence the main property that an ABA agent must satisfy in its operation is that indeed this follows these informed choices. This is the central *soundness* property of an ABA agent in that it follows the intended design as captured in the decision policies of its modules.

In this section we define such desirable properties and indicate how we can design ABA agents (in particular their IAC theories) that would satisfy them.

Property 1. An ABA agent such that for any state, $\langle V, \mathcal{D} \rangle$, of its operation, every decision $D \in \mathcal{D}$ holds in a maximal acceptable extension of the argumentation theory, $T(V)$, of the corresponding module grounded in the state V, (i.e. D is optimal w.r.t. the policy in its module in the world state V), is called a **strongly sound agent.**

A strongly sound agent is therefore one whose decisions are not only optimal at the time that they are taken but remain optimal at any subsequent situation where its view of the world may have changed. It is easy to see that we can

build such ABA agents by fixing their cautiousness at the highest level and designing their IAC to abandon decisions as soon as their commitment level falls below level 4 in the course of action and the passage of time. Indeed, let us choose the commitment level of a module's decisions to be given by the degree of acceptance of the decisions according to its (object level) expert policy theory as given in Definition 7. Then the high-level nature of the IAC theory allows us to specify, in the P_{Status} theory part of IAC, an argument: $[Arg(abandon, D, low)]$: $abandon(D)$ **if** $commitment\_level(D, V, C), C < 4$.

By giving, in the P_{Status} theory, to this argument higher-priority than any other argument (for keeping a decision) in P_{Status} we ensure that the IAC argumentation theory will always decide sceptically to abandon any decision when this is no longer preferred in the module's policy for choosing its decisions. In practice though in some applications this may be too strong to require as it may mean that decisions are abandoned too often. This can be mitigated, *e.g.*, by taking the cost induced by discarding this decision into account, or by requiring a weaker form of soundness where only some of the decisions are optimal throughout the operation of the agent. In particular, the higher level decisions in the "hierarchy" of modules, such as the goal decisions should remain optimal. Moreover, whenever any one of its goals is achieved (i.e. holds in the current state) then this should be optimal.

Property 2. An ABA agent such that for any state, $\langle V, \mathcal{D} \rangle$, of its operation, every goal decision, G, in $\mathcal{D}$ is acceptable in the state V, i.e. it holds in a maximal acceptable extension of the argumentation theory of the Goal Decision module grounded in the state V, is called a **sound agent**. Moreover, if whenever G holds in the current view of the world, V, the goal G is acceptable in the state V, then the agent is called a **sound achieving** agent.

Here we are assuming that once goals are achieved (as perceived by the agent in its world view) they are then immediately deleted from the state of the agent and that only goals that do not currently hold are added to the state. Achieved goals may later become suboptimal but this is beyond any reasonable requirement on the operation of an agent.

In effect all these properties of soundness are properties which require adaptability of the agent as it operates in an unknown environment. They require that the operation of the agent adapts to the new circumstances of the environment by changing its decisions accordingly. This high level of adaptability is facilitated in the ABA agents by the high level nature of their intra-agent control which allows them to recognize the changing status of decisions.

The above properties do not emphasize the overall internal coherency of the ABA agents as they are concerned with the individual internal decisions in each module. These individual choices need to be coherent with each other and give some overall sense to the agent's operations. This is given by the Motivations and Needs policy of the agent: the agent must operate in accordance to its current high-level desires and needs. We can therefore (re)formulate properties of soundness of the agent based on its motivations/desires.

Property 3. A **soundly motivated** ABA agent is an agent such that for any state, $\langle V, \mathcal{D} \rangle$, of its operation, and for every decision, D, in $\mathcal{D}$, D is acceptable in the state V with respect to the Motivations and Needs policy of the agent, whenever this policy is applicable to the corresponding module of D. In particular, its goal decisions in any state are always acceptable with respect to the Motivations and Needs policy of the agent.

Therefore a soundly motivated agent always operates according to the underlying motivations and needs policy that generates the agent's current desires. We can build such agents by suitably defining their IAC in a similar way to that of building sound agents, as shown above, where now instead of referring to the status of the decisions wrt object-level policy of the module we refer to the Motivations and Needs policy of the agent when this relates to the decision at hand. Indeed, we note that the soundly motivated property is essentially the only global consistency requirement that makes sense in an ABA agent, as there is no other global or explicit control of the agent.

6 Conclusions

The link between argumentation and multi-agent systems was originally viewed essentially as a way to manage the potentially conflicting knowledge bases of individual agents. With time this link has become much stronger covering several features of modern agency theories, *e.g.* negotiation, decision-making, communication. We have proposed an agent architecture uniformly based on argumentation with a highly modular structure. The focus is on a high-level architecture mainly concerned with managing the currently available best options for the agent's constituent tasks in a way that provides a coherent behaviour, with a focus of purpose, for the agent. This focus of purpose is governed to a certain extend by the agent's internal argumentation theory for its Motivations and Needs that gives the currently preferred high-level desires of the agent which in turn affect other decisions of the agent.

An important distinguishing characteristic of an ABA agent is that the agent's decisions are not rigid but rather they are decisions for currently preferred options or choices that its argumentation reasoning produces. These results of argumentation can be different under a different view of the world. This means that the agent is flexible and versatile in a changing environment, able to adapt graciously to changes in the agent's current situation, without the heavy need for an explicit mechanism of adaptation.

The aim of our work has been to present a high-level architecture based uniformly on argumentation which could then be used as a basis for developing such agents. This architecture and its argumentation basis does not depend critically on any specific argumentation framework but only requires some quite general properties of any such framework to be used. Different realizations can be developed by adopting anyone of the many concrete frameworks of argumentation that are now available, such as [21, 4, 11, 14, 2], particularly those which are preference based. Also aspects from different approaches to argumentation can be

exploited together within the ABA architecture. For example, the recent work of [10, 5] can be useful for the modular and distributed nature of the argumentation theories of the agent in its various modules. Moreover, the significant progress, over the recent years, in the study of the computational models of argumentation, e.g [11, 9, 17], can provide a platform for the practical construction of ABA agents. Nevertheless, our work constitutes a first step in the proposal to build agents uniformly based on argumentation. A proper validation of the proposed ABA architecture can only be achieved by developing specific applications with ABA agents and evaluating their performance both in terms of capturing desirable properties of the agents and the approach as a whole and in terms of its computational viability.

6.1 Acknowledgements

We would like to thank the organizers of the 2008 Dagstuhl "Perspectives Workshop: Theory and Practice of Argumentation Systems" where this work was started. In particular we would like to thank Juergen Dix and Paul Dunne for participating in the initial discussions of this work.

References

1. S. Airiau, L. Padham, S. Sardina, and S. Sen. Incorporating learning in bdi agents. In *Proceedings of the ALAMAS+ALAg Workshop*, May 2008.
2. T. J. M. Bench-Capon. Value-based argumentation frameworks. In *NMR*, pages 443–454, 2002.
3. E. Blanck and K. Atkinson. Dialogues that account for different perscpectives in collaborative argumentation. In *Proc. 8th Int. Joint Conf. on Autonomous Agents and Multiagent Systems*, pages 867–874, 2009.
4. A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.
5. G. Brewka and T. Eiter. Argumentation context systems: A framework for abstract group argumentation. In *LPNMR*, pages 44–57, 2009.
6. J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. van der Torre. The boid architecture: conflicts between beliefs, obligations, intentions and desires. In *AGENTS '01*, pages 9–16, New York, NY, USA, 2001.
7. Y. Dimopoulos, P. Moraitis, and L. Amgoud. Theoretical and computational properties of preference-based argumentation. In *ECAI*, pages 463–467, 2008.
8. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence Journal*, 77:321–357, 1995.
9. P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artif. Intell.*, 171(10-15):642–674, 2007.
10. P. M. Dung and P. M. Thang. Modular argumentation for modelling legal doctrines in common law of contract. *Artif. Intell. Law*, 17(3):167–182, 2009.
11. A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *TPLP*, 4(1-2):95–138, 2004.

12. A. C. Kakas, P. Mancarella, F. Sadri, K. Stathis, and F. Toni. Computational logic foundations of kgp agents. *J. Artif. Intell. Res. (JAIR)*, 33:285–348, 2008.
13. A. C. Kakas, R. Miller, and F. Toni. An argumentation framework of reasoning about actions and change. In *LPNMR*, pages 78–91, 1999.
14. A. C. Kakas and P. Moraitis. Argumentation based decision making for autonomous agents. In *AAMAS '03*, pages 883–890, 2003.
15. S. Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence Journal*, 2009.
16. M. Morge, K. Stathis, and L. Vercouter. Arguing over motivations within the v3a-architecture for self-adaptation. In *Proc. of the 1st International Conference on Agents and Artificial Intelligence (ICAART)*, pages 1–6, Porto, Portugal, 2009.
17. V. Noël and A. C. Kakas. Gorgias-c: Extending argumentation with constraint solving. In *LPNMR*, pages 535–541, 2009.
18. S. Parsons, C. Sierra, and N. R. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261—292, 1998.
19. S. Parsons, M. Wooldridge, and L. Amgoud. An analysis of formal inter-agent dialogues. In *AAMAS*, pages 394–401, 2002.
20. J. L. Pollock. Oscar: An architecture for generally intelligent agents. In *AGI*, pages 275–286, 2008.
21. H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *J. of Applied Non-Classical Logics*, 7:25–75, 1997.
22. A. S. Rao and M. P. Georgeff. BDI-agents: from theory to practice. In *Proceedings of the First International Conference on Multiagent Systems*, San Francisco, USA, 1995.
23. J. Sabater, C. Sierra, S. Parsons, and N. R. Jennings. Engineering executable agents using multi-context systems. *J. Log. Comput.*, 12(3):413–442, 2002.
24. M. C. Schut, M. Wooldridge, and S. Parsons. The theory and practice of intention reconsideration. *J. Exp. Theor. Artif. Intell.*, 16(4):261–293, 2004.
25. F. Toni. Argumentative agents. In *IMCSIT*, pages 223–229, 2010.
26. Q. B. Vo and N. Y. Foo. Reasoning about action: An argumentation - theoretic approach. *J. Artif. Intell. Res. (JAIR)*, 24:465–518, 2005.
27. M. Witkowski and K. Stathis. A dialectic architecture for computational autonomy. In *Agents and Computational Autonomy*, pages 261–274, 2003.
28. M. J. Wooldridge and A. Rao. *Foundations of rational agency.* Kluwer, 1999.