Argumentation-Based Reasoning in BDI Agents Using Toulmin's Model

Vágner de Oliveira Gabriel*†, Diana Francisca Adamatti†, Alison R. Panisson*,
Rafael H. Bordini* and Cleo Zanella Billa†
*School of Technology – Pontifical Catholic University of Rio Grande do Sul (PUCRS) – Porto Alegre – Brazil
†Center of Computational Science – Federal University of Rio Grande (FURG) – Rio Grande – Brazil
Emails: {vagner.gabriel, alison.panisson}@acad.pucrs.br, rafael.bordini@pucrs.br, {dianaadamatti, cleobilla}@furg.br

***Abstract*—The theory of argumentation spans several fields of knowledge, gaining significant space in the community of multiagent systems because it gives support for agents to reason about uncertain beliefs. This work describes the development of an argumentation-based inference architecture for BDI agents, which was developed based on Toulmin's model of argumentation. The philosopher Stephen Toulmin claimed that arguments typically consist of six parts: data, warrants, claim, backing, qualifiers, and rebuttals. Using the proposed architecture, an agent is able to create new beliefs based on available evidence and to justify such beliefs.**

***Index Terms*—Multiagent Systems, Argumentation-based Reasoning, Toulmin's Model.**

I. Introduction

In a society, people communicate to make decisions and solve problems in order to achieve goals. In agent societies it is not different: agents can communicate to solve problems, reduce conflicts, and inform facts to each other [1]. To improve such a communication process, it is desirable that an agent can argue about its decisions and beliefs.

Agents can be defined as virtual entities that act in an environment in which they interact and collaborate with other agents motivated by *goals* [2]. Agents can act in isolation or in groups, forming multiagent systems [3], [4]. Along the years, several architectures for multiagent systems were developed: [5] presented the rational architecture called BDI, which was based on the model proposed by [6] allowing agents to explicitly represent the following mental attitudes: *Beliefs*, *Desires*, and *Intentions*.

The theory of argumentation is studied in different fields of knowledge, as in Rhetoric, Law, Communication, Philosophy, and Artificial Intelligence (AI). In the AI area, argumentation has constantly evolved, being used to provide reconciliation between information and decision making. Some models of argumentation that are influential in AI are presented in [7]–[9].

In recent years, argumentation has received significant space in the multiagent community due to the possibility of providing an agent with the ability to reason about information available to ti, to reason about conflicting information, to reason about information acquired through perception of the environment, and to reason about information obtained from various agents through communication. In the literature, it is possible to find much work of great importance in the multiagent community involving argumentation [9]–[14].

In multiagent systems, argumentation can be divided into two main lines of research: (i) Argument-based reasoning, which is used on incomplete, conflicting, or uncertain information, where arguments for and against a conclusion (e.g., a belief or a goal) are constructed and compared; (ii) Argument-based dialogues where agents interact by exchanging arguments (e.g., providing justifications to support claims).

This paper presents an argumentation-based inference architecture for BDI agents which aims to provide components that support agent reasoning about beliefs. The architecture proposed in this work has its structure defined in accordance with the main ideas introduced in [15]. Stephen Toulmin presented an alternative model of argument rather than the traditional one; the alternative model has as a key point the structural analysis of an argument.

Toulmin's model defines six components when constructing an argument: data, which are facts and evidence used to prove an argument; warrants, which are general statements that serve as bridges between the claim and the data; qualifiers, which are statements that limit the strength of the argument or statements that propose the conditions under which the argument is true; rebbutals, which are counter-arguments or statements indicating circumstances when the general argument does not hold true; and backing, which are statements that serve to support the warrants (i.e., arguments that do not necessarily prove the main point being argued, but which do prove that the warrants are true).

The architecture proposed in this paper allows agents to build new beliefs based on the six components of Toulmin's model and also to justify a certain belief because it can be decomposed into these components.

In the next section, we present Toulmin's model and how it was adapted to the BDI model. Then, we show the theoretical model of the architecture for argumentation-based reasoning in BDI agents. The next section shows how we implement the architecture in AgentSpeak. We then present a case study and discuss our experiments using the architecture.

II. Toulmin's Model

In 1958, Stephen Toulmin presented an alternative model of argumentation to the traditional syllogism [15], specifying

978-1-5386-8023-0/18/$31.00 ©2018 IEEE
DOI 10.1109/BRACIS.2018.00072

the structure of an argument and detailing various components of an argument. That model is not restricted to a discussion involving two or more individuals, it can be used for individual reasoning, where an individual draws conclusions from information based on the knowledge available to them [16].

According to [15], there are facts that support a claim, provided that claim was not made in an irresponsible manner. On the event that a claim is challenged, the individual can present the facts that support the claim. An example that illustrates a challenge of a claim may be as follows. An individual claims that "Paul is Brazilian." This individual may be questioned about this statement: "What do you have to keep up with that claim?". So, the individual needs to argue about what they relied on to come up with such a claim. "I have the knowledge that Paul was born in Rio Grande do Sul." This is the individual's belief that serves as basis for the claim.

From this point, we will abbreviate data as D, which represents the facts used to support a claim. The claim will be abbreviate as C, which is the claim that the individual establishes as true. Following the example presented, a new question may arise, asking for a relationship between the data and the claim. The next step is to present propositions that ground the data as support for the claim. This propositions are called warrants (W), which are used as implicit assumptions that act as a bridge between data and claim. In this case, the example would look like this: the knowledge that the individual has about Paul being born in Rio Grande do Sul gives the right to discard the suggestion that Paul is not Brazilian, because of the warrant that "A man born in Rio Grande do Sul is Brazilian".

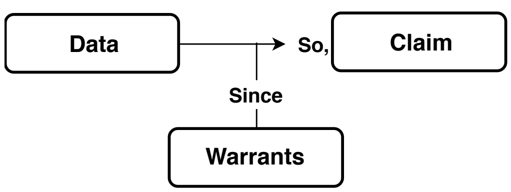

Fig. 1. Toulmin's model with components Data, Warrant, and Claim [15].

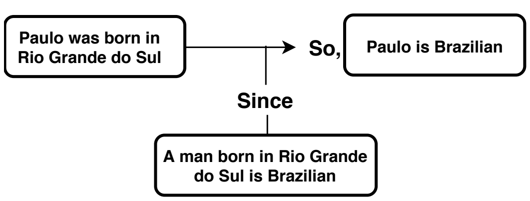

Fig. 2. Example of Toulmin's model with components Data, Warrant, and Claim.

In Figure 2, the arrow symbolizes the relationship between the data and the claim, where the data supports the claim. The warrant under the arrow can be seen as a bridge from the data to the claim.

There may be different types of warrants, which provide different degrees of strength for a claim. Thus, it is necessary to qualify the confidence of the claim based on the pair (data, warrant). Also, there may be facts that challenge the claim. So Toulmin's model has two components, called qualifier (Q) and rebuttal (R) to represent the complete structure of an argument. The qualifier stands next to the claim, indicating the strength of the claim, and it is a relation of the warrant and the rebuttal components.

After presenting the data, the warrants, the qualifiers, and the rebuttals, the challenger may still question the warrants. In the example, it was assumed that a person born in Rio Grande do Sul is Brazilian, but the challenger of the claim may question: "What is the source of this warrant?". Therefore, another component, the Backing (B), is added to the model. The Backing component represents the source of a warrant (i.e., a law, a specialist, etc.). Figures 3 and 4 show the complete version of Toulmin's model.

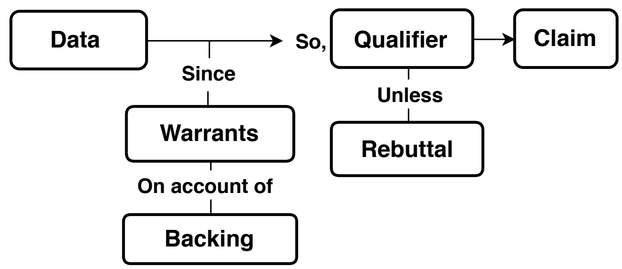

Fig. 3. Toulmin's model with all six components.

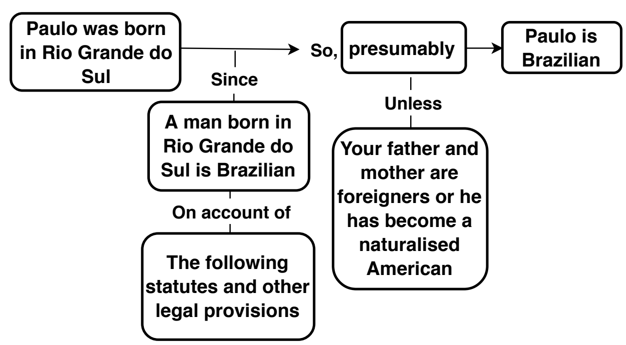

Fig. 4. Example of Toulmin's model with all six components.

III. An Architecture for Argumentation-based Reasoning in BDI Agents based on Toulmin's Model

The proposed architecture is composed of six components: Data (D), Claim (C), Warrant (W), Backing (B), Rebuttal (R) and Qualifier (Q). These components are used as a basis for quantifying the authenticity of a claim. The architecture components are described in Table I.

Based on the components of Toulmin's model, we develop an architecture for argumentation-based inference in BDI agents. Figure 5 shows the adapted components of Toulmin's model included in the PRS (Procedural Reasoning System)

TABLE I
ARCHITECTURE COMPONENTS.

Component	Description
Data	Beliefs that the agent may have at the time of its creation, or acquired from the environment, or received through communication with other agents, or through communication with human users.
Warrant	Beliefs that support the claim.
Rebuttal	Beliefs that disprove the claim.
Backing	Legal documents supporting the warrants and rebuttals.
Claim	A new belief generated by an agent's reasoning about data beliefs, warrants, and rebuttals.
Qualifier	A weight that quantifies the support that a claim has.

architecture. In Figure 5, it can be seen that the Data, Warrants, Rebuttals and the Claim components are beliefs while the qualify function is located in the plan library.

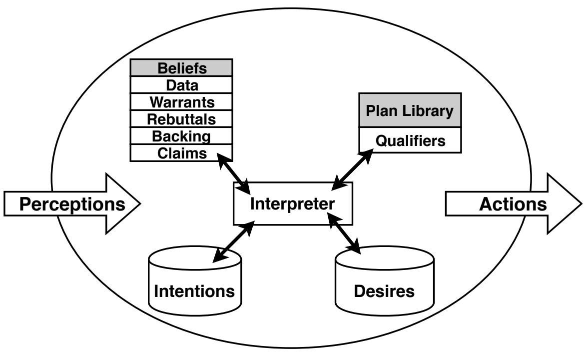

Fig. 5. Toulmin's Model in BDI Agents (Toulmin Architecture).

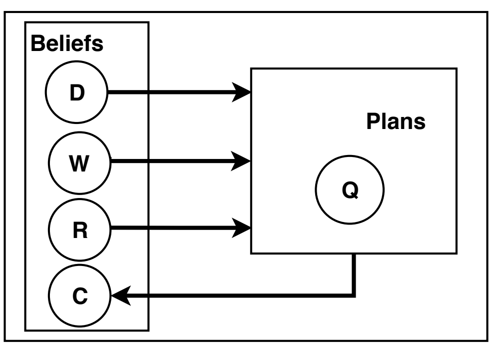

Fig. 6. Toulmin's Model in BDI Agents (Toulmin Reasoning).

Figure 6 illustrates the functioning of the developed architecture. Initially, there are Data, Warrants and Rebuttals, which are beliefs that the agent acquired through communication or perception of the environment. These beliefs are used in agent's plans. The plan uses a function called qualify, which receives the sets of data, warrant, and rebuttal beliefs and returns a value that is the confidence that the claim has based on the agent's beliefs. Finally, the plan creates a claim with its respective qualifier which becomes a new belief in the agent's belief base.

```
/* beliefs */
data(data1).
data(data2).
data(data3).

warrant(warrant1, weight1).
warrant(warrant2, weight2).
warrant(warrant3, weight3).

rebuttal(rebuttal1, weight1).
rebuttal(rebuttal2, weight2).
rebuttal(rebuttal3, weight3).
```

Fig. 7. Representation of Beliefs in AgentSpeak.

A. *The Qualify Function*

The qualify function q aims to receive data, warrant, and rebuttal beliefs, and with those beliefs generate a confidence value for the claim. This qualify function can be implemented according to the needs of the user, adapting to various domains. The signature of the qualify function is:

$$q : \mathscr{P}(D) \times \mathscr{P}(W) \times \mathscr{P}(R) \longrightarrow \mathbb{Q}$$

where Q is a user-defined set of qualifiers

IV. ARCHITECTURE IMPLEMENTATION

To demonstrate how the developed architecture works, the theoretical model presented in the previous section was implemented in the Jason platform [17]. Jason is a platform that enables the development of multiagent systems using Java and AgentSpeak. To implement the architecture in AgentSpeak is straightforward, as that language is based on the BDI model.

A. *Beliefs*

In the proposed architecture, Data, Warrants, and Rebuttals are beliefs. Thus, we implement these components as follows:

- data(data1).
- warrant(warrant1, weight1).
- rebuttal(rebuttal1, weight1).

Data beliefs represent information that the agent received at the time of its creation or acquired through perception of the environment or interactions with other agents through communication.

Warrant beliefs are information that combined with data will generate a new belief (claim). Each warrant has a weight, which ranges from 0.0 to 1.0. This value increases according to the level of confidence that the claim will have based on the Data beliefs.

The Rebuttal beliefs are information that reduces the confidence of the claim (e.g., Paul is *not* Brazilian). Each rebuttal also has a weight, which varies between 0.0 and 1.0 and increases according to the level of certainty that the data will discourage the claim.

B. *Plans*

For the implementation of the architecture, we need an AgentSpeak plan that based on the agent's data, warrants, and rebuttals beliefs generates a new belief with its respective qualifier. Within this plan there is a qualify function that receives as inputs the beliefs, warrants, and rebuttals and returns a confidence qualifier. After the end of the qualify function process, the plan generates a new belief with the confidence qualifier.

As mentioned previously (subsection IV-A), a plan needs a function that searches for all relevant beliefs (with the pre-defined predicates we use), since in the architecture there will be several data, warrant, and rebuttal beliefs. The Jason platform has a pre-defined internal action called `.findall`, which performs the task of fetching terms that match a particular pattern and inserting them all into a list. The plan implemented in the AgentSpeak language can be seen in Figure 8. Note that this is a preliminary implementation aimed primarily at testing the ideas we incorporated from Toulmin's model; one possible future direction is to use separate Jason modules for each claim the agent may need to make.

```
/* Plans */
+!plan:true
<- .findall(X, data(X), ListData);
   .findall(Y, warrant(Y), ListWarrant);
   .findall(Z, rebuttal(Z), ListRebuttal);
   toulmin.qualifier(ListData,  ListWarrant,
    ListRebuttal, Qualifier);
   +claim("plan", Qualifier).
```

Fig. 8. Skeleton of a Plan for a Claim in AgentSpeak.

C. The Qualify Function

In our architecture, the qualify function is a function that must return the level of confidence of a claim. It can be computed in several ways, depending on the problem. In this work, the weights that qualify each warrant and each rebuttal are values between 0.0 and 1.0. The value of the weights increases according to the level of certainty that the warrants and rebuttals influence the claim that will be generated.

The qualify function was defined to calculate the Sum of the Weights of the Warrants (SWW) and to calculate the Sum of the Weights of the Rebuttals (SWR) that refute the input data. In order to support or refute a claim, some data must match a warrant or a rebuttal, for example, the data belief: "Paul was born in Rio Grande do Sul" supports the claim because there is a warrant: "Paul was born in Rio Grande do Sul, 1.0". There may be any number of warrants and rebuttals, but only the weights of warrants and rebuttals that match a data will be considered.

However, if there is any entry data contained in the warrants and/or rebuttals with maximum weight (1.0), the qualify function will consider only the warrant and/or rebuttal with that weight:

$$q_{Final} = \begin{cases} 1.0 & \text{if } \exists\ i\ .\ WW_i = 1.0 \\ 0 & \text{if } \exists\ i\ .\ RW_i = 1.0 \\ 0.5 & \text{if } \exists\ i\ .\ WW_i = 1.0 \text{ and } \exists\ i\ .\ RW_i = 1.0 \end{cases}$$

If there is any input data that matches a warrant with value 1.0, the weight that qualifies the claim will be 1.0, generating a symbolic qualifier of "Certainly yes". We explain the symbolic qualifiers below. If there is any data that matches a rebuttal with weight 1.0, the weight that qualifies the claim will be 0, generating a symbolic qualifier of "Certainly not" for the claim. However, if there is data matching both warrants and rebuttals with values 1.0, this is an inconsistency, generating a symbolic qualifier of "Maybe" for the claim, because the agent cannot determine its confidence.

Otherwise, equations 1 and 2 will be computed.

$$SWW = \sum_{i=1}^{nvw} WW_i \tag{1}$$

$$SWR = \sum_{i=1}^{nvr} WR_i \tag{2}$$

In Equation 1, *nvw* stands for the number of valid warrants and (WW_i) is the weight of each valid warrant (valid here means that it matches the data). Equation 2 does the same as Equation 1, but summing over the data matching the list of rebuttals. In Equation 2, *nvr* stands for the number of valid rebuttals and WR_i is the weight of each valid rebuttal.

Subsequent to the computing of the sums, the values are normalized. The qualifier is computed subtracting $SWR_{normalized}$ from $SWW_{normalized}$ generating a value as shown in Equation 3.

$$q = SWW_{normalized} - SWR_{normalized} \tag{3}$$

After calculating Equation 3, q_{Final} is computed using equation 4, which is the final confidence level of the claim.

$$q_{Final} = \begin{cases} \mid q \mid & \text{if } SWW > SWR \\ 1 - \mid q \mid & \text{if } SWR > SWW \end{cases} \tag{4}$$

The q_{Final} function above qualifies the confidence level of a claim. It is a value between 0.0 and 1.0, which is transformed into a symbolic qualifier accordingly to the following rules:

- $[0, 0.2) \rightarrow$ "Certainly not.";
- $[0.2, 0.4) \rightarrow$ "Hardly.";
- $[0.4, 0.6] \rightarrow$ "Maybe.";
- $(0.6, 0.8] \rightarrow$ "Presumably.";
- $(0.8, 1.0] \rightarrow$ "Certainly.".

V. Case Study

This case study was adapted from [15] and aims to answer whether a person named Paul is Brazilian or not. In the study, the warrants act as supporters of the claim that "*Paul is Brazilian*" and the rebuttals act to reduce the confidence in that claim. Table II presents the beliefs for data, warrants, and rebuttals used in this case study.

Figure 9 shows the information contained in Table II in the implemented version.

In Table II, there are five warrants and two rebuttals matching the data. Therefore, they are the only values used to generate the qualifier.

The first step in the qualifier function is to check for data that matches a maximum weight warrant and/or rebuttal (1.0).

TABLE II
CASE STUDY – PAUL'S NATIONALITY.

Component	Information
Data	Paul lives in Rio Grande do Sul.
	Paul's parents are Brazilian.
	Paul's wife was born in Rio Grande do Sul.
	Paul has a sister who lives in London.
	Paul has a Brazilian son.
	Paul works in Rio Grande do Sul.
	Paul studied in London.
Warrants	Paul was born in Rio Grande do Sul, 1.0.
	Paul lives in Rio Grande do Sul, 0.2.
	Paul's parents are Brazilian, 0.2.
	Paul has a Brazilian son, 0.3.
	Paul works in Rio Grande do Sul, 0.2.
	Paul's wife was born in Rio Grande do Sul, 0.1.
	Paul has a sister who lives in Rio Grande do Sul, 0.2.
Rebuttals	Paul was born in London, 1.0.
	Paul lives in London, 0.2.
	Paul's wife was born in London, 0.1.
	Paul's has a sister who lives in London, 0.2.
	Paul's studied in London, 0.1.

```
/* Belief */
data("Paul lives in Rio Grande do Sul").
data("Paul's parents are Brazilian").
data("Paul's wife was born in Rio Grande
           do Sul").
data("Paul has a sister who lives in London").
data("Paul has a Brazilian son").
data("Paul works in Rio Grande do Sul").
data("Paul studied in London").

warrant("Paul was born in Rio Grande do Sul",
          1.0).
warrant("Paul lives in Rio Grande do Sul",
          0.2).
warrant("Paul's parents are Brazilian", 0.2).
warrant("Paul has a Brazilian son", 0.3).
warrant("Paul works in Rio Grande do Sul", 0.2).
warrant("Paul's wife was born in Rio Grande do
          Sul", 0.1).
warrant("Paul has a sister who lives in Rio
          Grande do Sul", 0.2).

rebuttal("Paul was born in London", 1.0).
rebuttal("Paul lives in London", 0.2).
rebuttal("Paul's wife was born in London",
            0.1).
rebuttal("Paul's has a sister who lives in
            London", 0.2).
rebuttal("Paul's studied in London", 0.1).

/* Goal */
!paulBrazilian.

/* Plan */
+!paulBrazilian : true
 <- .findall(X, data(X), ListData);
    .findall(Y, warrant(Y), ListWarrant);
    .findall(Z, rebuttal(Z), ListRebuttal);
    toulmin.qualifier(ListData, ListWarrant,
     ListRebuttal, Qualifier);
    +claim(Qualifier, "Paul is brazilian").

```

Fig. 9. Case Study – Nationality in AgentSpeak.

In this case, there is none, so the qualify function computes the sum of the valid warrants and rebuttalls:

$$SWW = (0.2 + 0.2 + 0.3 + 0.2 + 0.1) = 1.0$$
$$SWR = (0.2 + 0.1) = 0.3$$

Subsequently, Equation 3 is applied using the normalized values of SWW and SWR:

$$q = (1.0 - 0.3) = 0.7$$

After the calculation of Equation 3, q_{Final} is computed, which is the final confidence level of the claim.

$$SWW > SWR \rightarrow q_{Final} = \mid 0.7 \mid = 0.7$$

In this example, the qualifier has value 0.7. This value is between 0.6 and 0.8, thus the symbolic qualifier for the claim is "Presumably".

If we have the same warrants and rebuttals, but different data, the qualifier would be different, and the symbolic qualifier for the claim could be different too. For example:

- If the agent believes that "Paul was born in Rio Grande do Sul", the symbolic qualifier would be "Certainly", since the data "Paul was born in Rio Grande do Sul" matches a maximum weight warrant.
- If the agent believes that "Paul was born in London", the symbolic qualifier would be "Certainly not", since data "Paul was born in London" matches a maximum weight rebuttal.
- If the agents believes that "Paul's wife was born in London" rather than "Paul's wife was born in Rio Grande do Sul", the symbolic qualifier would be "Maybe", since the sum of the weights of warrants and rebuttals would be very similar.

VI. RELATED WORK

This work proposes a new architecture for argumentation-based reasoning in BDI agents. Different approaches to argumentation-based reasoning can be found in the literature, for example [9], [18]–[28]. None of them is based on Toulmin's Model of Argumentation. This model is very powerful because it decomposes an argument into various components, facilitating the ways an agent can explain its reasoning.

The work presented in [24] uses a practical approach for the construction of argumentative BDI agents and is based on the abstract structure of [7], which is used to develop a module to be integrated into the Jason platform. The author has developed a module that is decoupled from the traditional BDI model since it only operates on the beliefs of agents and there is no interference in the execution of plans, goal adoption, or agent commitments. The main objective of that paper is to allow the agents to query the argumentation module to obtain suggestions for attacks or justifications based on arguments for accepted or rejected beliefs.

In [25], [26], the authors have developed an argumentation-based reasoning mechanism in an agent-oriented programming language, which has its properties defined according to the BDI model. The argument-based mechanism developed by the author has its foundation in the mechanism of defeasible logic, and enables agents to reason about uncertainty and to argue to support their claims by exchanging messages with other agents. The authors used the AgentSpeak language to implement their approach. Also, they claim that, in their approach, agents can query the existence of arguments and their acceptability in their own reasoning during the execution of a plan and during a dialogue.

In [27], the authors presented the implementation of an argumentation system for participatory management of protected environmental areas, specifically modeling a park management agent. This implementation was based on the BDI architecture, using the AgentSpeak language. The authors modeled a system of argumentation through different layers of knowledge base and the relation of attacks between arguments, in order to generate a basis for selecting better viable arguments. The authors present a case study where the proposed architecture was tested, modeling a park management agent in a serious game for participatory management of protected areas. The park management agent aims to make decisions about conservation types by examining and arguing about the situation and concerns of the protected area.

The difference of the architecture we propose here in relation to the work in [24]–[27] is the use of Toulmin's model to make inferences in BDI agents, so an argument can be decomposed into components, thereby facilitating the justification of particular claims. This architecture divides a claim into components, which allow for a deeper analysis of how the reasoning was performed and, consequently, how to argue about it.

VII. Conclusion

The central objective of this study was the development of an argumentation-based architecture for reasoning in BDI agents. The architecture developed in this work had its foundation on the model presented by Toulmin, which provides an alternative to traditional argumentation models. The architecture developed enables not only the generation of claims, but it also qualifies the level of confidence of a claim.

The advantage of using the developed architecture is to allow a modular analysis of an argument, making it possible to analyze each component of the architecture separately. In this way, it becomes possible to check which argument had a determinant weight in a conclusion. Our architecture also adapts to different scenarios, as we standardize the structure of the architecture according to the foundations laid out by Toulmin. The architecture also allows the implementation of different qualify functions, so it can adapt to different domains.

References

[1] E. Oliva, P. McBurney, and A. Omicini, "Co-argumentation artifact for agent societies," in *International Workshop on Argumentation in Multi-Agent Systems*. Springer, 2007, pp. 31–46.

[2] J. Ferber and L. Gasser, *Intelligence artificielle distribue*. France: Tutorial notes of the 11th conference on expert systems and their applications, 1991.

[3] S. O. Rezende, *Sistemas Inteligentes: fundamentos e aplicações*, Barueri, São Paulo, 2003.

[4] M. Wooldridge, *An introduction to multiagent systems*. John Wiley & Sons, 2009.

[5] A. S. Rao, M. P. Georgeff *et al.*, "Bdi agents: From theory to practice." in *ICMAS*, vol. 95, 1995, pp. 312–319.

[6] M. Bratman, *Intention, plans, and practical reason*, 1987.

[7] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," *Artificial intelligence*, vol. 77, no. 2, pp. 321–357, 1995.

[8] H. Prakken and G. Sartor, "Argument-based extended logic programming with defeasible priorities," *Journal of applied non-classical logics*, vol. 7, no. 1-2, pp. 25–75, 1997.

[9] I. Rahwan, S. D. Ramchurn, N. R. Jennings, P. Mcburney, S. Parsons, and L. Sonenberg, "Argumentation-based negotiation," *The Knowledge Engineering Review*, vol. 18, no. 4, pp. 343–375, 2003.

[10] S. Kraus, K. Sycara, and A. Evenchik, "Reaching agreements through argumentation: a logical model and implementation," *Artificial Intelligence*, vol. 104, no. 1-2, pp. 1–69, 1998.

[11] C. Sierra, N. Jennings, P. Noriega, and S. Parsons, "A framework for argumentation-based negotiation," *Intelligent agents IV agent theories, architectures, and languages*, pp. 177–192, 1998.

[12] L. Amgoud, S. Parsons, and N. Maudet, "Arguments, dialogue, and negotiation," in *Proceedings of the 14th European Conference on Artificial Intelligence*. IOS Press, 2000, pp. 338–342.

[13] L. Amgoud, N. Maudet, and S. Parsons, "Modelling dialogues using argumentation," in *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*. IEEE, 2000, pp. 31–38.

[14] L. Amgoud and C. Cayrol, "A reasoning model based on the production of acceptable arguments," *Annals of Mathematics and Artificial Intelligence*, vol. 34, no. 1, pp. 197–215, 2002.

[15] S. E. Toulmin, *The uses of argument*. Cambridge university press, 2003.

[16] D. Hitchcock, "Good reasoning on the toulmin model," *Argumentation*, vol. 19, no. 3, pp. 373–391, 2005.

[17] R. H. Bordini, J. F. Hübner, and M. Wooldridge, *Programming multi-agent systems in AgentSpeak using Jason*. John Wiley & Sons, 2007.

[18] A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni, "An abstract, argumentation-theoretic approach to default reasoning," *Artificial intelligence*, vol. 93, no. 1-2, pp. 63–101, 1997.

[19] A. Kakas and P. Moraitis, "Argumentation based decision making for autonomous agents," in *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM, 2003, pp. 883–890.

[20] K. Atkinson, T. Bench-Capon, and P. Mcburney, "A dialogue game protocol for multi-agent argument over proposals for action," *Autonomous Agents and Multi-Agent Systems*, vol. 11, no. 2, pp. 153–171, 2005.

[21] K. Atkinson, T. Bench-Capon, and P. McBurney, "Generating intentions through argumentation," in *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. ACM, 2005, pp. 1261–1262.

[22] K. Atkinson and T. Bench-Capon, "Practical reasoning as presumptive argumentation using action based alternating transition systems," *Artificial Intelligence*, vol. 171, no. 10-15, pp. 855–874, 2007.

[23] N. D. Rotstein, A. J. García, and G. R. Simari, "Defeasible argumentation support for an extended bdi architecture," in *International Workshop on Argumentation in Multi-Agent Systems*. Springer, 2007, pp. 145–163.

[24] T. Berariu, "An argumentation framework for bdi agents," in *Intelligent Distributed Computing VII*. Springer, 2014, pp. 343–354.

[25] A. R. Panisson, F. Meneguzzi, R. Vieira, and R. H. Bordini, "An approach for argumentation-based reasoning using defeasible logic in multi-agent programming languages," in *11th International Workshop on Argumentation in Multiagent Systems*, 2014.

[26] A. R. Panisson and R. H. Bordini, "Knowledge representation for argumentation in agent-oriented programming languages," in *5th Brazilian Conference on Intelligent Systems (BRACIS)*. 2016, pp. 13–18.

[27] P. E. Velmovitsky, J.-P. Briot, M. Viana, and C. J. P. de Lucena, "Practical reasoning in an argumentation-based decision bdi agent: a case study for participatory management of protected areas," in *29th International Conference on Software Engineering & Knowledge Engineering (SEKE'2017)*. SEKE/Knowledge Systems Institute, 2017, pp. 527–530.

[28] S. Parsons, K. Atkinson, K. Haigh, K. Levitt, P. M. J. Rowe, M. P. Singh, and E. Sklar, "Argument schemes for reasoning about trust," *Proceedings of Computational Models of Argument (COMMA)*, vol. 245, p. 430, 2012.