Computational Models of Argument
H. Prakken et al. (Eds.)

doi:10.3233/FAIA200538

ArgAgent: A Simulator of Goal Processing for Argumentative Agents

Henrique M. R. JASINSKI [a], Mariela MORVELI-ESPINOZA [a] and Cesar A. TACLA [a]
[a] *Graduate Program in Electrical and Computer Engineering (CPGEI), Federal University of Technology - Paraná (UTFPR), Curitiba, Brazil*

1. Introduction

Since multiagent systems are intrinsically distributed, debugging and explaining their behaviour poses a especial challenge. In [1] a new abstract model for intelligent agents is presented, called Belief-Based Goal Processing (BBGP), which is different from the Belief-Desire-Intention (BDI) model [2] mainly due to the "goal processing" responsible for choosing which goals should be pursued. In [3] and [4] a computational formalization of the BBGP is presented which uses argumentation for the goal processing reasoning. We call this model Argumentative-BBGP. The developed simulator is a tool that allows an Argumentative-BBGP agent to be executed and to inspect its decision process.

ArgArgent[1] was developed using Java. Two libraries, and its dependencies, from the TweetyProject [5] were used. The first-order logic module was used to represent the basic elements of the model, such as the beliefs and goals. The ASPIC argumentation module was used in the goal processing for non-monotonic reasoning. The focus of the simulator is primarily the "goal processing". Each goal may be in one of the following states: **active**, **pursuable**, **chosen**, **executive**, **completed**, or **canceled**, in that order, where necessarily a goal must have attained the previous states, with the exception of canceled state, which can be achieved from any state. The goal processing comprises four well defined stages: I) **activation**, which instantiates goals based on the agents current beliefs; II) **evaluation**, which identifies and evaluates obstacles for pursuing active goals; III) **deliberation**, which identifies the associated plans for each pursuable goal, evaluates conflicts among pursuable, chosen, and executive goals, and determines which pursuable goals should become chosen; and IV) **checking**, which evaluates whether the conditions to execute the plan for every chosen goal hold.

To start the simulation, a file containing the agent's initial beliefs, rules, the set of plans, and the preference total order on the goals is required. Each rule must be either strict or defeasible. It is also possible to load a perception file containing the perception itself and the simulation cycle in which they occur. Once the simulation begins, it is possible to inspect the current agent beliefs and the perceptions that it receives at a given cycle. It is also possible to inspect the goals memory, which describes when a given

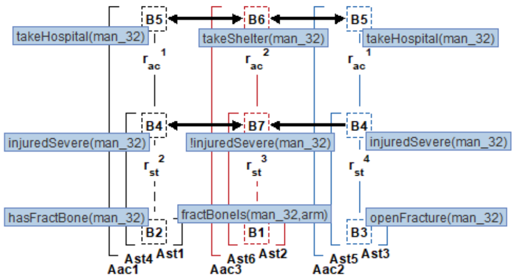
[1] ArgAgent can be found at www.github.com/henriquermonteiro/BBGP-Agent-Simulator.

Acknowledgement This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 - and the Centro Nacional de Pesquisa - Brasil (CNPq).

```
  Beliefs:
=> openFracture(man_32)
=> typeHolder(none)
=> beOperative(me)
=> supportWeight(man_32)
=> hasFractBone(man_32)
=> newSupply(bed)
=> fractBoneIs(man_32, arm)
=> askedForHelp(p2, p6)
=> !available(bed, Y)

  Standard Rules:
openFracture(X) -> injuredSevere(X)
fractBoneIs(X, arm) => !injuredSevere(X)
hasFractBone(X) => injuredSevere(X)
newSupply(X) -> available(X,Y)
```

(a) Belief inspector



(b) Goal memory inspector

Figure 1. (a) shows the beliefs and some rules of the rescue agent, where '−>' and '=>' represent strict and defeasible rules, respectively. (b) shows the interactive diagram of a goal memory entry. Beliefs, goals, and rules receive an identifier to make the diagram more readable, but a tooltip with the full description is available. Arguments receive their own identifier as well, and each bracket indicates an argument. The arrows represent attacks among arguments. The argument in blue is the selected one. Arguments in red are the rejected ones, and the ones in black are accepted arguments that defend the selected one.

goal attained a status and the beliefs that supported such decision. Figure 1 shows an example of a rescue agent, which must decide whether to send *man_32* to the hospital or to the shelter. Figure 1(b) shows the reasoning process which led to the decision of taking *man_32* to the hospital ('Aac2'), since he had an open fracture, which in turn is a severe injure. Such decision took place because the rule $openFracture(x) \rightarrow injuredSevere(x)$ is strict.

We plan to improve the simulator by implementing the mechanism for changing a goal state towards a previous state and cancellation. Our aim is to create an agent model capable of explaining in more details his decision process compared with similar approaches.

References

[1] Castelfranchi, C., Paglieri, F. The role of beliefs in goal dynamics: prolegomena to a constructive theory of intentions. Synthese, v. 155, n. 2, p. 237–263, 2007.

[2] Rao, A., Georgeff, M. BDI Agents: From Theory to Practice. Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95), p. 312-319, 1995.

[3] Morveli-Espinoza, M., Possebom, A. T., Puyol-Gruart, J., Tacla, C. A. Argumentation-based intention formation process. Journal of the National University of Colombia (DYNA), 86(208), pp. 82-91, January - March, 2019.

[4] Morveli-Espinoza, M., Nieves, J. C., Possebom, A., Puyol-Gruart, J. and Tacla, C. A. An argumentation-based approach for identifying and dealing with incompatibilities among procedural goals. International Journal of Approximate Reasoning, v. 105, p. 1–26. Elsevier Inc, 2019.

[5] Thimm, M. Tweety: A comprehensive collection of Java libraries for logical aspects of artificial intelligence and knowledge representation. Proceedings of the Fourteenth International Conference on Principles of Knowledge Representation and Reasoning, p. 528–537, 2014.