# Self-Explaining Agents in Virtual Training

Maaike Harbers[1,2], Karel van den Bosch[2], and John-Jules Meyer[1]

[1] Utrecht University, P.O.Box 80.089, 3508 TB Utrecht, The Netherlands
[2] TNO, Kampweg 5, 3796 DE Soesterberg, The Netherlands
maaike@cs.uu.nl,jj@cs.uu.nl,karel.vandenbosch@tno.nl

**Abstract.** Virtual training systems are increasingly used for the training of complex, dynamic tasks. To give trainees the opportunity to train autonomously, intelligent agents are used to generate the behavior of the virtual players in the training scenario. For effective training however, trainees should be supported in the reflection phase of the training as well. Therefore, we propose to use self-explaining agents, which are able to generate and explain their own behavior. The explanations aim to give a trainee insight into other players' perspectives, such as their perception of the world and the motivations for their actions, and thus facilitate learning. Our project investigates the possibilities of self-explaining agents in virtual training systems, and the effects on learning.

## 1 Introduction

Virtual training is used to train people for complex, dynamic tasks in which fast decision making is required, e.g. crisis management, military missions or fire-fighting. In typical virtual training, a trainee has to accomplish a given mission and therefore he has to interact with other virtual players, e.g. team-members, opponents, or neutral participants. Currently, in most virtual training these are controlled by other trainees or instructors. However, using intelligent agents instead of humans gives trainees more flexibility to train where and whenever they want, and it reduces costs. Fire-fighters could for example train during a night shift, when they spend most of their time waiting for an alarm.

Intelligent agents can only (partly) replace humans if they are able to generate believable behavior, which might be complex. Moreover, trainees should be supported to reflect on the training because that promotes learning [13]. Reflection could be provoked by providing (the possibility to request for) explanations about the virtual players' behavior, which can give a trainee insight into their perspectives. Such a facility requires intelligent agents that are able to explain their actions, so called self-explaining agents.

In this paper, we present a PhD project issuing the topic of self-explaining agents in virtual training. In section 2, we discuss some related work, and in section 3 we give a formulation of our research question. Then, we provide a more detailed discussion on our approach and the results achieved so far in section 4. We end the paper with a conclusion and an outline of future research in section 5.

## 2 Related work

A lot of research has been done on intelligent tutoring systems (ITS) [11, 10], which is a topic closely related to self-explaining agents. ITSs teach students how to solve a problem or execute a task by giving explanations during and after task execution. ITSs have been successfully designed for the training of well-structured skills and tasks such as programming or mathematics. In contrast, tasks that are being trained in virtual training systems are usually real world, complex and dynamic. The space of possible actions of a trainee is large, and often there is no single 'right' way to accomplish a task. So instead of explanations that give hints and recipes of what is to be done as provided by ITSs, explanations in virtual training should give insight into the processes in the training scenarios. Trainees can use these to make sense of the situation and construct a picture of what is going on themselves, and thus reflect on their own performances.

A few proposals for self-explaining agents in virtual training systems have been made. The first called Debrief [9], which has been implemented as part of a fighter pilot simulation and allows trainees to ask an explanation about any of the artificial fighter pilot's actions. To generate an answer, Debrief modifies the recalled situation repeatedly and systematically, and observes the effects on the agent's decisions. With the observations, Debrief determines what factors were responsible for 'causing' the decisions. However, Debrief derives what *must have been* the agent's underlying beliefs for an action, but sometimes an action has several possible explanations. If (some of) the agent's reasoning steps would be made explicit instead of derived from observable behavior, the actual reasons for executing an action could be given.

A more recently developed account of self-explaining agents is the XAI explanation component [14]. The XAI system has been incorporated into a simulation-based training for commanding a light infantry company. After a training session, trainees can select a time and an entity, and ask questions about the entity's state. However, the questions involve the entity's physical state, e.g. its location or health, but not its mental state.

A second version of the XAI system [4, 1] was developed to overcome the shortcomings of the first; it claims to support domain independency, modularity, and the ability to explain the motivations behind entities' actions. This second XAI system is applicable to different simulation-based training systems, and for the generation of explanations it depends on information that is made available by the simulation. Most simulations however do not represent agents' goals, and preconditions and effects of actions, and thus still no explanations of agents' reasons can be given.

## 3 Research question

The previous section showed that ITSs usually support the learning of well-structured tasks, in which it is clear what actions are right and wrong. In the virtual training systems we focus on however, training tasks can be achieved in many different ways and require another type of feedback than provided by ITSs. Self-explaining agents might be a good alternative, but we believe that the existing accounts lack some crucial aspects. They either just give explanations about the agent's physical state, or they *derive* information about the agent's mental state from its behavior or the simulation. We believe that an agent's behavior can best be explained by its actual underlying motivations, i.e. information about its mental state, and that the explanation component thus should have direct access to this information and not on their effects. To solve the shortcomings in the current solutions, the PhD project presented in this paper addresses the following research question:

*How can we develop self-explaining agents and how can they be applied in virtual systems to support training?*

The question is two-fold, the first part is a technical question, and the second part focuses more on educational aspects. In the remainder of this paper we discuss our methodology, and the results achieved so far.

## 4 Our approach

To obtain direct access to an agent's reasons for executing an action, we believe that behavior explanation should be connected to the generation of behavior. The deliberation steps that are taken to generate an action can also be best used to explain that action, and when these deliberation steps are understandable, the explanations should be as well. To obtain understandable deliberation steps, the agent's reasoning elements should have some level of abstraction. For instance, the description "an agent is opening a door" is more useful for understanding its behavior than "an agent is moving object x from position (x1,y1,z1) to (x2,y2,z2)". We have chosen to use a BDI-based (beliefs desires intentions) approach [12], so that our agents reason with concepts such as beliefs, desires and plans, and also provide explanations in these terms.

We have chosen for the BDI approach because it matches the way humans give explanations. Humans adopt a certain 'stance' or 'mode of construal' for explaining and predicting phenomena, and different stances are chosen to explain different phenomena [7]. Dennett for example distinguishes the mechanical, the design, and the intentional stance [3]. The intentional stance considers entities as having beliefs, desires and other mental contents, and fits most natural to explain the behavior of humans or virtual characters that behave like humans. To understand the behavior of agents, it only matters whether they behave *as if they had* beliefs and desires. However, agents that have to generate understandable explanations based on their deliberation should also have actual beliefs and desires and reason with them.

The BDI approach defines an agent's reasoning elements, but it does not tell how actions can be generated from an agent's goals and beliefs, i.e. how planning works. For an account of planning, we have looked at the GPGP (generalized partial global planning) approach [8]. The GPGP approach is a framework for the coordination of small teams of agents and makes use of task structures. TAEMS (task analysis, environment modeling and simulation) is the language used to represent these task structures. The underlying model of the GPGP approach can be represented conceptually as an extended AND/OR goal tree in which task are decomposed into subtasks which in turn are decomposed, etc. The leaves of the tree are primitive (non-decomposable) actions.

Conform the GPGP approach, we structure the possible goals, plan and actions of an agent in a task hierarchy. The task at the top of the hierarchy is an agent's goal, the subtasks possible plans for reaching that goal, and the leaves are the agent's actions. Consequently, for each of the agent's goals, a task hierarchy is defined. The actions that an agent executes to achieve a goal depend on three aspects (explained in the next paragraph): its beliefs, the nature of the task-subtask relation, and its preferences.

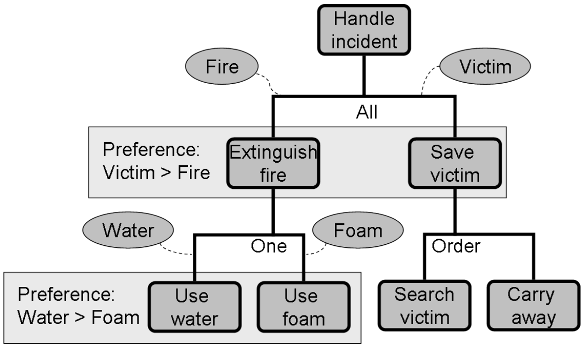

**Fig. 1.** The goal, plans and actions (boxes), and beliefs (circles) of a fire-fighting agent

First, the GPGP model is designed for a team of agents, but we take a single agent perspective. Therefore, in our model the beliefs of a single agent can be added to the task hierarchy, to form the conditions that determine which tasks can possibly be executed. Second, three task-subtask relations can be distinguished. A task can be executed when:

- **All** subtasks are executed
- **One** subtask is executed
- All subtasks are executed in a specific **Order**

Third, the agent's preferences determine the order in which subtasks are executed in an All-relation, and which subtask is executed in an One-condition.

Figure 1 shows a the model of a simple fire-fighting agent. The agents main goal is to handle the incident, and it has several plans available to achieve this goal. Its current beliefs determine how the agent 'walks through' the hierarchy. The first step is to choose between saving a victim (if the agent beliefs that there is an actual victim), or extinguishing a fire (if the agent beliefs that there is a fire and no victim because saving victims is preferred over extinguishing fires). For saving a victim, it first has to search the victim and then carry it away. For extinguishing a fire it can either use water or foam, dependent on its beliefs about the availability of water and foam.

The same agent model can also be used for the generation of explanations about the agent's actions. The actions that are the result of an agent's deliberation process can be explained by the beliefs, goals and reasoning steps that were in involved in the process. For instance, extinguishing a fire with foam could be explained as follows.

```
I wanted to handle the incident,
and I believed there was a fire and no victim,
therefore I wanted to extinguish the fire,
and I believed there was foam and no water,
therefore I used foam
```

Such explanations can become quite long, which is not desired [7]. Therefore, the most informative elements in the explanations have to be selected, e.g. 'I used foam because there was no water'.

For the implementation of our agent model it is required that the agent's reasoning elements and its deliberation steps can be explicitly represented in the programming language. Second, an agent needs to have access to this information. We have chosen to implement our model in the agent programming language 2APL [2]. 2APL is a BDI-based programming language, so the goals, plans and beliefs of an agent can explicitly be represented. Moreover, a 2APL agent is capable of introspection into its own beliefs and goals. For more details on the agent model and its implementation see [5, 6].

## 5 Conclusion

We argued that virtual training of complex and dynamic tasks requires intelligent agents that can provide explanations for their behavior in such a fashion that it helps trainees to improve their understanding. So far, we have developed an agent model capable of the generation and explanation of behavior, and we have made an implementation of the model.

We are currently reviewing literature on cognitive behavior research to determine which information people find most useful in explanations. Based on the outcome of this study we want to develop filters which select the most useful information out of longer explanations. Furthermore, we want to extend the agent model with factors like emotions, personality or social contracts, which may also influence an agent's behavior. Explanations referring to these properties may help trainees to become more sensitive and understanding to them. The next step will be to connect the self-explaining agent to an existing virtual training system, and perform user experiments.

We believe that our approach can create a learning tool that is currently not existing, and which fulfills the requirements of autonomous training of complex and dynamic training tasks. Our goal in this project is to demonstrate that the self-explaining agents we are developing deliver appropriate and useful explanations, leading to improved learning.

**Acknowledgements** This research has been supported by the GATE project, funded by the Netherlands Organization for Scientific Research (NWO) and the Netherlands ICT Research and Innovation Authority (ICT Regie).

## References

1. M. Core, T. Traum, H. Lane, W. Swartout, J. Gratch, and M. van Lent. Teaching negotiation skills through practice and reflection with virtual humans. *Simulation*, 82(11), 2006.
2. M. Dastani. 2APL: a practical agent programming language. *Autonomous Agents and Multi-agent Systems*, 16(3):214–248, 2008.
3. D. Dennett. *The Intentional Stance.* MIT Press, 1987.
4. D. Gomboc, S. Solomon, M. G. Core, H. C. Lane, and M. van Lent. Design recommendations to support automated explanation and tutoring. In *Proc. of the 14th Conf. on Behavior Representation in Modeling and Simulation*, Universal City, CA., 2005.
5. M. Harbers, v. d. Bosch, K., F. Dignum, and J. Meyer. A cognitive model for the generation and explanation of behavior in virtual training systems. In *Proc. of Exact 2008*, Patras, Greece, Forthcoming.
6. M. Harbers, F. Dignum, v. d. Bosch, K., and J. Meyer. Explaining simulations through self-explaining agents. In *Proc. of EPOS 2008*, Lisbon, Portugal, Forthcoming.
7. F. Keil. Explanation and understanding. *Annual Reviews Psychology*, 57:227–254, 2006.
8. V. Lesser, K. Decker, N. Carver, A. Garvey, D. Neiman, M. Nagendra Prasad, and T. Wagner. Evolution of the GPGP/TAEMS domain-independent coordination framework. *Autonomous agents and muli-agent systems*, 9:87–143, 2004.
9. W. Lewis Johnson. Agents that learn to explain themselves. In *Proc. of the 12th Nat. Conf. on Artificial Intelligence*, pages 1257–1263, 1994.
10. T. Murray. Authoring intelligent tutoring systems: An analysis of the state of the art. *Internat. Journal of Artificial Intelligence in Education*, (10):98–129, 1999.
11. M. Polson and J. Richardson. *Foundations of Intelligent Tutoring Systems.* Lawrence Erlbaum Associates, Inc., Mahwah, NJ, 1988.
12. A. Rao and M. Georgeff. Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proc. of the 2nd Internat. Conf. on Principles of Knowledge Representation and Reasoning*, pages 473–484, San Mateo, CA, USA, 1991. Morgan Kaufmann publishers Inc.
13. D. Schon. *Educating the Reflective Practitioner.* Jossey-Bass, San Francisco, 1987.
14. M. Van Lent, W. Fisher, and M. Mancuso. An explainable artificial intelligence system for small-unit tactical behavior. In *Proc. of IAAA 2004*, Menlo Park, CA, 2004. AAAI Press.