The Virtual Apprentice

Weizi Li and Jan M. Allbeck

Laboratory for Games and Intelligent Animation
George Mason University
4400 University Drive, MSN 4A5
Fairfax, VA 22030
{wlia,jallbeck}@gmu.edu

Abstract. Over the past couple of decades, virtual humans have been attracting more and more attention. Many applications including, video games, movies, and various training and tutoring systems have benefited from work in this area. While the visual quality of virtual agents has improved dramatically, their intelligence and socialization still needs improvement. In this paper, we present work towards endowing agents with social roles and exploiting Explanation-Based Learning (EBL) to enable them to acquire additional, contextual behaviors from other agents. These virtual humans are capable of learning and applying role related actions from multiple agents and only adopt behaviors that have been explained to them, meaning that their definition of a role may be a subset from one or more agents. This results in emergent behaviors in heterogeneous populations.

Keywords: Virtual Humans, Social Roles, Explanation-Based Learning

1 Introduction

Virtual humans have increasingly attracted attention over the last decade. Many applications including games, movies, urban and transportation planning systems, and training and tutoring simulators are prospering due in part to this burgeoning technology. Observing the potential, researchers from various disciplines have invested tremendous effort to improve these artificial lives' visual quality and life-like behaviors. While many impressive strides have been made, we believe virtual humans can be further elevated in two respects: incorporation of social roles and inclusion of learning and evolving abilities.

We argue that it's important to include social roles into virtual humans for several reasons. First, roles can better organize agent behaviors and demonstrate agent internal attributes such as goals and duties. In the real world, we usually have multiple roles, and our behaviors, goals and obligations are heavily associated with each of them. Thus, for virtual agents, roles are an ideal tool for governing various behaviors and their incentives. Secondly, for longer duration simulations in which agents are continuously learning and evolving in order to perform long-term tasks, roles are needed to improve consistency, believability, and reasonableness. Third, certain domain-specific simulations, for example military and medical scenarios, can be only realized with the presence of social roles such as commanders, soldiers, civilians, doctors and patients. In the virtual human and animation research community, though roles have been assigned to characters in many applications and various planning algorithms have been developed to control agent behaviors, more studies are needed to show how roles could be adopted and the interrelationship between roles and behaviors.

The ability to learn has been successfully applied to software agents, robots and virtual characters, particularly in applications for communicating with real humans. However, learning between virtual characters still needs further exploration. For example, we can imagine Non-Player Characters (NPCs) in a game learning strategies for combating player actions from each other, creating a reasonable and increasingly challenging evolution of game play. Furthermore, the behavior selection mechanisms for virtual characters are often hand crafted and static. The knowledge base of the agents is assumed to be complete, resulting in agents that lack the ability to demonstrate emergent behaviors. This limits their use in the current and future applications. Virtual humans capable of developing contextual behaviors will facilitate longer, more compelling simulations and games.

The purpose of this work is trying to partially fill the gap and advance behavioral animation by endowing virtual humans with social roles and the ability to learn. Specifically, we address role adopting phenomenon via learning by observation and explanation. We exploit an Explanation-Based Learning (EBL) mechanism, allowing the agent, through their observation and other agents' explanations, to acquire new knowledge and concepts based on prior knowledge, and eventually be capable of adopting new roles. In order to better actualize our idea, we also introduce semantics to our virtual world by organizing all objects and their features into an ontology. Additionally, for illustrating the effectiveness of our approach, we describe an example in which the resulting agents demonstrate more reasonable interactions and behaviors more consistent with their environment and roles.

2 Related Work

In order to create believable agent behaviors, numerous efforts have invested in developing sophisticated behavior selection mechanisms and simulating an agent's decision-making process. Some researchers have explored computational models such as decision-networks [33] and fuzzy logic [15]. Others have demonstrated the use of various social-psychology factors such as in [24, 19] or the BDI architecture [26]. In addition, a great deal of work has addressed this problem using cognitive approaches, for example [12, 23, 13]. Similar to ours, there exist several works that exploit semantics to facilitate agent behaviors. For example, [11] annotates the virtual environment with information to support navigation of the agents and their interactions with the objects. Chang et al [5] and Kao et al [17] integrate semantics into the agent planning and reasoning processes.

Though generating significant results, for the most part, these research efforts assume the agent knowledge base is complete. In other words, the number of agent behaviors is fixed. Thus agents are not only prohibited from learning and evolving, but also cannot demonstrate emergent behaviors.

While few above mentioned works incorporate social roles, a virtual train station with pedestrian performing different roles is described in [28]. Grimaldo et al [14] simulate a virtual university bar with agents acting in two roles: waiter and customer. Pelechano et al. include roles among other factors to simulate an evacuation scenario [24]. In our previous work [18], we have also simulated social roles and explored the idea of role switching. However, we did not include an ability to learn and evolve individual definitions of roles. Another group of work includes using social roles to communicate with real humans (e.g. [16, 32]) in serving training and tutoring scenarios [29]. Although these works have assigned agents social roles, the agents are still lacking an ability to learn and few utilize roles as a tool to organize agent internal attributes and behaviors.

In contrast, there are several works that have endowed their virtual characters with a learning ability. For example, Blumberg et al [3] integrated learning activity to a synthetic character. Orkin et al designed a restaurant game [22] in which they collect behavior and dialog from real human players and later apply them to virtual characters to enhance a gaming experience. The idea of learning from observation in [8] is similar to ours, but their approach addresses interactions between software agents and real world experts. Still other work addresses learning activity entirely within a virtual or simulated environment. Cohen et al [6] built a learning baby with sensorimotor interactions in a simulated environment and Conde et al [7] use reinforcement learning to allow characters to find their path to goal locations. However, the learning activities in these works are not among virtual characters but between characters and the environment. To summarize, having virtual humans learn from real humans often requires a lot of effort from the human participants. Also, enabling virtual characters to learn from each other in addition to the environment will further enhance their behaviors. A virtual human population that evolves more autonomously is both more natural and less demanding of simulation authors.

3 Semantic Virtual World

Semantics can be very helpful when constructing operable virtual worlds. They can be used to better organize knowledge of the environment such as objects and their features. They can also facilitate agent-object and agent-agent interactions by endowing agents with the ability to retrieve corresponding features and information efficiently. In addition, by separating environment characteristics from agent stories, object entities become scenario-independent and can be easily applied to other environments without massive modifications.

We have adopted an ontology to hold virtual world semantics. The ontology consists of hierarchical classes, properties, and relations between instances of the classes. An instance is the child of at least one class and is described by properties which link it to various values, including numbers, strings, and other instances and classes. All objects and their features are stored and updated in the ontology. The object features include geometry, color, status, and spatial information. A small part of our ontology is shown in Fig. 1. Later in the paper we will show how the ontology greatly assists agents in reasoning about the virtual world and learning new knowledge and concepts.

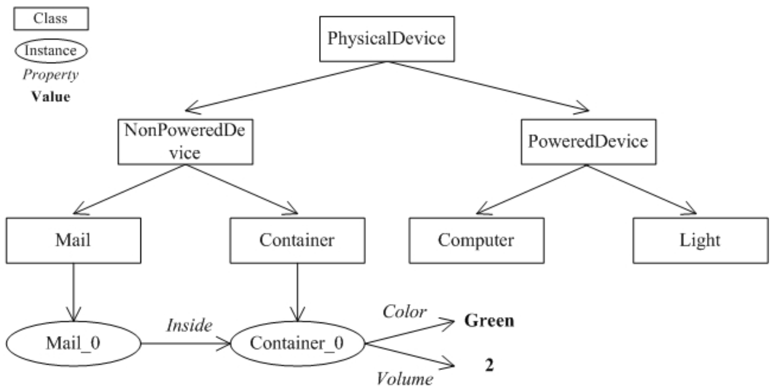

Fig. 1. Partial ontology of object entities

4 Intelligent Social Agents

In this section, we will first provide a definition of social roles extracted from socio-psychology literature and then explain in more detail our learning strategy.

4.1 Social Roles

According to [1], a role is *the rights, obligations, and expected behavior patterns associated with a particular social status.* In Stark's textbook, *Sociology*, he indicates that roles can be achieved or assigned by someone else [30]. Furthermore, they can be semi-permanent, such as having an occupation, or they can be transitory, such as being a patient. Ellenson's work [10] points out that each person could play a number of roles, or in other terms, engage in a *role set*. With these definitions and descriptions, and also by taking into account discussions from other social-psychology work [2, 20], we conclude that roles are patterns of behaviors for given situations or circumstances. They can be achieved, assigned and abandoned, having various durations and are often associated with social relationships. Furthermore, multiple roles can be possessed by an individual at the same time. Given this summation, agent will switch from one role to another by performing characteristic behaviors of the latter role. For example, a *Trainee* could switch to *Administrator* by performing its feature behaviors such as *Post flyers* and *Organize professor mail* which could be freely defined by users. For more extensive discussion about role switching phenomenon, we refer readers to our previous paper [18].

What's more, people can have their own definition and expectations of a single role. For example, being a professor to some may include both *research* and *teaching*, while for others conducting either *research* or *teaching* solely is considered enough to have that role. Because each role consists of several characteristic behaviors, obligations, and duties and each individual can have their own definition and expectations for a certain role, our society is colorful and diverse. To create more virtual human heterogeneity, we also allow our agents to have different definitions of various roles.

Lastly, roles can be influenced and constrained by many factors, such as biology or genetics. For instance, a female is unlikely to take on the role of father, and some athletes and musicians seem genetically predisposed to excel at those roles. This implies that certain roles, or more specifically, certain behaviors have physical and intellectual prerequisites. In this work, we assume that agents attain a set of prerequisites for performing elementary actions such as talking, nodding, carrying, picking up, and also possess the ability to learn.

4.2 Learning Strategy

Nearly since the birth of the computer, various learning methods have been developed and used to solve real world problems effectively and efficiently. While this powerful tool, learning, has been successfully utilized in building software agents, robots and more, its usage in simulating interactions between autonomous virtual agents residing in virtual worlds has not been fully explored. Currently many applications using virtual humans, such as video games and training simulators, adopt scripted behaviors and lots of "if - then" rules. This not only inhibits the virtual characters ability to learn and evolve, it also makes the configuration of simulations laborious, since for each different scenario, a mound of extra rules need to be designed and included. In this work, we are trying to partially resolve this problem by equipping our virtual agents with an ability to learn and allowing them to learn from each other. To proceed, we would like to point out that our goal of incorporating a learning mechanism differs from more conventional applications. Traditionally, as we have mentioned, learning methods are used to solve certain tasks more efficiently and accurately. Here, since we are simulating virtual humans and they are expected to behave, reason and learn like real humans, our goal in including a learning method is to generate more reasonable simulations and enhance behavioral animation. This distinction also explains why we adopt a specific learning method rather than just copying knowledge from one agent to another.

The learning phenomenon of real humans is very complex and still under discussion in terms of its exact form and process. Nevertheless, it is widely believed that it involves certain approaches such as explanation-based learning, analogical learning, instance-based learning and reinforcement learning. Also we know that under certain conditions a particular learning method is favored over the others. Given this problem is extremely sophisticated, we are not trying to simulate all aspects of real human learning activity but concentrate on learning by observation and explanation. To achieve this, we have adopted Explanation-Based Learning (EBL). EBL is an analytical learning method. Based on prior knowledge, observation, explanation, and expanded information provided by training examples, new knowledge and concepts can be learned [9, 21, 31, 27]. While we acknowledge not all skills and concepts can be acquired through observation and explanation, in many cases we do obtain knowledge in this fashion. For example, imagine you are a trainee who is going to work in an office environment. At the beginning, you probably need to learn various duties from observation and your supervisor's explanation. In other social settings, for instance, traveling to a different country, when you are learning the local culture and manners, most likely the learning method is also observation and explanation. For acquiring this kind of knowledge and concepts, other learning methods seem less plausible. To be specific, we do not have abundant examples needed for inductive learning methods such as decision tree learning and neural networks or possess many similar examples we can compare with in order to carry out instance-based learning or face situations fulfilled with probabilities that Bayesian networks could manage or attain direct and/or indirect feedback as a training source for reinforcement learning. With these considerations and after taking several other learning methods into account, we find EBL is the most plausible and effective approach.

In general, EBL includes the following components (for a more thorough discussion, we refer readers to [9, 21]): *Goal concept*, a target concept with a set of relevant features; *Training example*, a typical positive example of a concept to be learned; *Domain theory*, prior knowledge which can be used to analyze or explain why the training example could satisfy the goal concept; And finally a *Learned Rule*. As one may notice, one of the keys to this approach is prior knowledge assignment. We need to determine what kind of knowledge should be given to our agents in order to achieve generality and scenario-independence. To address this, we have found some psychology studies showing that babies are born with physical and spatial reasoning [4] and language acquiring abilities [25]. Even though there are no conclusions about which abilities are innate, given that we are simulating normal intellectual and physical level adult-like agents, we believe it is reasonable to give them at least following three categories of base knowledge while still preserving the generality:

- Color: $Red, Green, Blue, Yellow, Cyan \ldots$
- Spatial Relationship: $Inside, Outside, Above, Below, \ldots$
- Common Object Type (lowest level of our ontology class): $Mail, Container, Computer, \ldots$

Finally, we need to mention that one premise that needs to be met in order to successfully perform EBL is that all prior knowledge has to be correct. This premise is to ensure that all further inferences drawn would also be correct. However, we believe in virtual humans simulation this criteria can be loosen since it is reasonable and natural for one to have false knowledge and later draw

Fig. 2. A school environment

false inferences. Actually this might accentuate the imperfect nature of human behaviors in our virtual humans.

5 Implementation and Example

In this section, we will detail our implementation of the learning and role adopting process through an extended example. Most commonly when using EBL, agent learning is through observation and explanation of a series of actions performed by real world experts. Since we are aiming to create a purely autonomous world, we adapt the term observation and explanation to indicate such behaviors occurring between virtual humans. One trainee can observe other agents' actions and these agents can explain their current action series. In this fashion, the trainee learn new knowledge and concepts.

As an example scenario, we have created the school environment shown in Fig. 2. This example includes three agents. One takes the role of *Trainee*, while other two become an *Administrator* and a *Housekeeper*. The goal of this simulation is to teach the trainee several duties associated with being an administrator and a housekeeper such that he will eventually be capable of adopting these two roles.

In this particular example, we form the duties of an *Administrator* as *Organize professor mail*, *Post flyers* and *Fill paper for office equipments* while a *Housekeeper*'s duties include *Check classroom* and *Water plant*. In order to successfully carry out these duties, the trainee must first learn several new concepts. For example, the place for storing a professor's mail needs to be known when performing *Organize professor mail*. The condition of a plant needs to be considered before performing *Water plant*. Here we use the former case to illustrate the learning process. Assume in this scenario we have two professors, namely *ProfA* and *ProfB*, and the trainee is learning from his trainer, the current administrator, where *ProfA*'s mail is. In this context, we put the learning task in EBL form as following:

- Goal Concept: $MailToProfA(x)$
- Training Example: A positive example, $MailToProfA(Obj1)$
  $Inside(Obj1, Office\_1)$
  $Inside(Obj1, Obj2)$
  $Type(Obj1, Mail)$
  $Type(Obj2, Container)$
  $Color(Obj1, White)$
  $Color(Obj2, Red)$
  ...
- Domain Theory:
  $MailToProfA(x) \leftarrow Location(x, Office\_1) \wedge Inside(x, y) \wedge Type(x, Mail) \wedge Type(y, Container) \wedge Color(y, Red)$
  $Location(x, Office\_1) \leftarrow Inside(x, Office\_1)$
  ...
- Learned Rule:
  $MailToProfA(x) \leftarrow Inside(x, Office\_1) \wedge Inside(x, y) \wedge Type(x, Mail) \wedge Type(y, Container) \wedge Color(y, Red)$

The final learned rule states "*Mail* x is for *ProfA* if x is inside *Office_1* and also inside y which is a *Container* and has the color *Red*". With this newly learned knowledge, the trainee can perform behaviors such as "Transfer *ProfA*'s mail to his container" and "Retrieve *ProfA*'s mail from his container".

The detailed implementation of this learning task is as following. First of all, we have two actions *Observe* and *Explain* associated with the trainee and administrator, respectively. Then, we put the positive example and the domain theory into a database and assign the positive example with a boolean value initially set to 0 indicating the current training status. Once the administrator starts to *Explain* and the trainee starts to *Observe*, an underlying recursive algorithm will begin. In each iteration, the algorithm will attempt to match and prove each concept in the domain theory by using facts in the positive example. The whole procedure will continue until the goal concept is proved. To be specific, for above example, when the procedure begins, the algorithm will first try to prove the concept $MailToProfA(Obj1)$. However, this will fail since the concept $Location(x, Office\_1)$ is not in prior knowledge (i.e. Color, Spatial Relationship and Common Object Type). Then the algorithm will continue to try to prove next concept $Location(x, Office\_1)$ and this time it will succeed because the fact $Inside(x, Office\_1)$ of the positive example is in the prior knowledge. After $Location(x, Office\_1)$ has been proved, the goal concept $MailToProfA(Obj1)$ will also be proved and the whole procedure will complete. At this point, the training status of the positive example switches from 0 to 1 indicating the *Explain* process and also the *Observe* process of the positive example are over. When these two processes are finished, the last step, "generalization" will begin. This step essentially replaces instances in proved positive example with variables. In general, since all the instances are organized in an ontology as we mentioned in Section 3, this replacement is simply climbing the object hierarchy. However, this procedure is subject to one piece of prior knowledge which is the Common Object Type. In the beginning of "generalization", the proved positive example would have following form: $MailToProfA(Obj1) \leftarrow Inside(Obj1, Office\_1) \wedge Inside(Obj1, Obj2) \wedge Type(Obj1, File) \wedge Type(Obj2, Container) \wedge Color(Obj2, Red)$, here since the $Obj1$ and $Obj2$ both have a type specified, *Mail* and *Container*, these two instances can climb object hierarchy only to their types. In contrast, if an instance in some rules does not have a type specified then it can climb the object hierarchy all the way to "PhysicalDevice" according to Fig. 1. With this, the whole learning process is considered complete and the knowledge has been added to the knowledge base of the trainee. From this example, we can also see an advantage of EBL, which is it filters non-relevant object features when forming the final learned rule, such as $Color(Obj1, White)$. This is similar to real world cases, where an object can have multiple features, but we only need to know some features for some tasks. Other concepts can be learned in a similar vein, for simplicity we only list the final learned rules:

- $MailToProfB(x) \leftarrow Inside(x, Office\_1) \wedge Inside(x, y) \wedge Type(x, Mail) \wedge Type(y, Container) \wedge Color(y, Green)$
- $ReadyForTransfer(x) \leftarrow Inside(x, Hallway\_1) \wedge Above(x, y) \wedge Type(y, Table) \wedge Color(y, Cyan)$
- $ReadyForPost(x) \leftarrow Inside(x, Hallway\_1) \wedge Above(x, y) \wedge Type(y, Table) \wedge Color(y, Yellow)$
- $ItemLeftInClassroom(x) \leftarrow Inside(x, Classroom) \wedge \neg Type(x, WhiteBoard) \wedge \neg Type(x, LectureDesk) \wedge \neg Type(x, StudentChair)$
- $NeedWatering(x) \leftarrow Type(x, Plant) \wedge Color(x, Yellow)$

Our approach is convenient and efficient. While this example had only a few rules, more can be added to the system just by providing positive examples and the corresponding domain theories in the database. Although the underlying algorithm is recursive, processing only applies to single examples and their domain theory which is succinct and independent. Given this, including more rules, tasks, objects, and roles does not dramatically increase the computational cost.

In addition, as we discussed in Section 4.1, every individual can have their own definition of a specific role, which contributes variety and diversity to our society. Here, we apply this feature for the purpose of generating heterogeneous virtual humans. To provide an illustration, in our example, *Trainee* will learn all duties of being an *Administrator* except *Fill paper for office equipment.* Alternatively, the trainee can learn partial duties from multiple administrators, but not an entire set from either. This will result in the trainee having a definition of *Administrator* that is consistent with, but different from others in the world. An agent's definition of a role can evolve over time as more tasks are observed and explained. The actions of the agent while in that role will then also evolve to correspond with the changing role definition. Both the learning and performing procedures are demonstrating in Fig. 3 and Fig. 4, respectively.

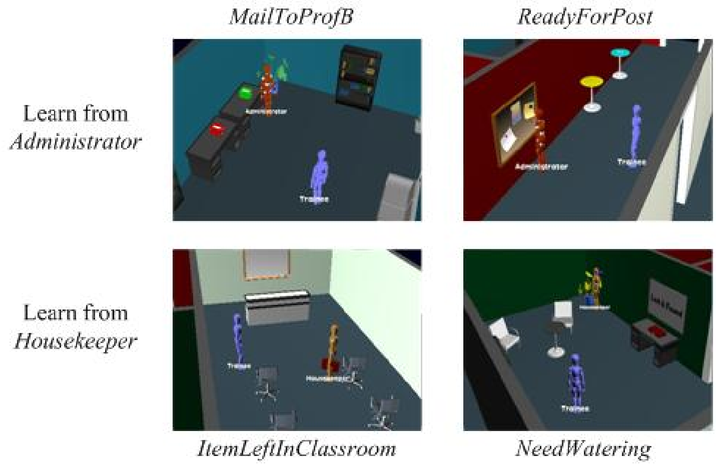

Fig. 3. Trainee learned concepts: *MailToProfB*, *ReadyForPost*, *ItemLeftInClassroom* and *NeedWatering*.

6 Conclusion and Future Work

Our ultimate goal is to be able to simulate populations of virtual humans over extended periods of time with reasonable behaviors that are appropriate to the context and evolve as the agents gain knowledge and experience just as they do in the real world. In this paper we have presented a method that uses roles and Explanation-Based Learning (EBL) to organize the agent behaviors and enable emergent behaviors. Furthermore, our approach is designed such that the learning is general and scenario independent. Agents could learn definitions of roles in one scenario and apply them in completely different scenarios. While the behaviors would be contextually reasonable and fitting in the new scenarios, they may not be exactly what the author has in mind for them. Often methods that increase the autonomy of the agents also decrease control over them. With our method, the learned rules for a role are stored in a database and could simply be deleted if they are not desired for new scenarios. One could also imagine simple interactive supervised learning techniques to eliminate undesired behaviors. Alternative learning techniques might also be used to enhance agent behaviors in other situations. For example, could an established learning technique be used to create emerging interpersonal relationships?

Because agents can learn partial definitions of roles and from multiple agents, heterogeneous populations evolve. Unfortunately, this means that conflicts can also arise. What if an agent is being taught conflicting behaviors for a role? One housekeeper explains that items left in classrooms should be put in lost and found, while another housekeeper explains that that can be kept and taken home. Which explanation should be used? Certainly a person's own individual differences including morals would have an impact, but another consideration is the status level of those involved. One agent might out rank the other. The status

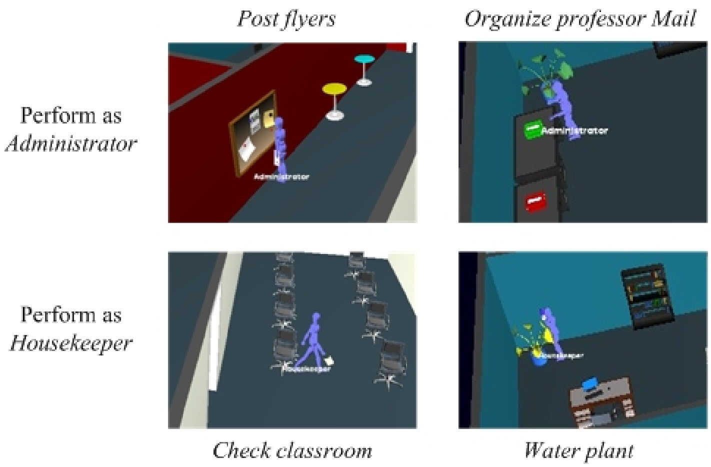

Fig. 4. Trainee adopts the role *Administrator* and *Housekeeper* and is performing the corresponding duties.

relationships between agents can be stored in a hierarchy and referenced to decide such conflicts. Such a hierarchy might also be used to distribute tasks when there are multiple agents with the same role. In addition, there exists certain complex task which consists of several sub-tasks. For learning and performing this kind of task, another database field could be added to indicate the learning and performing order of sub-tasks. Also, since the knowledge base of each agent is separated from other's, agents can even have different orders to learn and perform a complex task.

References

1. Webster's College Dictionary. Random House (1991)
2. Biddle, B.J.: Role Theory: Concepts and Research. Krieger Pub Co (1979)
3. Blumberg, B., Downie, M., Ivanov, Y., Berlin, M., Johnson, M.P., Tomlinson, B.: Integrated learning for interactive synthetic characters. In: Proceedings of the 2002 ACM SIGGRAPH Conference. pp. 417–426. ACM (2002)
4. Carey, S., Spelke, E.: Domain-specfic knowledge and conceptual change. In Hirschfeld, L. A., and Gelman, S. A., eds., Mapping the Mind. Cambridge University Press (1994)
5. Chang, P., Chien, Y.H., Kao, E., Soo, V.W.: A knowledge-based scenario framework to support intelligent planning characters. In: Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., Rist, T. (eds.) Intelligent Virtual Agents. LNCS, vol. 3661, pp. 134–145. Springer Berlin / Heidelberg (2005)
6. Cohen, P.R., Atkin, M.S., Oates, T., Beal, C.R.: Neo: learning conceptual knowledge by sensorimotor interaction with an environment. In: Proceedings of the first international conference on Autonomous agents. pp. 170–177. AGENTS '97 (1997)

7. Conde, T., Thalmann, D.: Learnable behavioural model for autonomous virtual agents: low-level learning. In: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems. pp. 89–96. AAMAS ’06 (2006)
8. Costa, P., Botelho, L.: Learning by observation in software agents. In: Proceedings of the 4th International Conference on Agents and Artificial Intelligence (ICAART) (2012)
9. Dejong, G., Mooney, R.: Explanation-based learning: An alternative view. Machine Learning 1, 145–176 (1986)
10. Ellenson, A.: Human Relations. Prentice Hall College Div; 2 edition (1982)
11. Farenc, N., Boulic, R., Thalmann, D.: An informed environment dedicated to the simulation of virtual humans in urban context. Computer Graphics Forum 18(3), 309–318 (1999)
12. Funge, J., Tu, X., Terzopoulos, D.: Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. In: Proceedings of the 1999 ACM SIGGRAPH Conference. pp. 29–38. SIGGRAPH ’99 (1999)
13. Goertzel, B., Pitt, J., Wigmore, J., Geisweiller, N., Cai, Z., Lian, R., Huang, D., Yu, G.: Cognitive synergy between procedural and declarative learning in the control of animated and robotic agents using the opencogprime agi architecture. In: Proceedings of the 25th AAAI National Conference on Artificial Intelligence. AAAI’11, AAAI Press (2011)
14. Grimaldo, F., Lozano, M., Barber, F., Vigueras, G.: Simulating socially intelligent agents in semantic virtual environments. Knowl. Eng. Rev. 23(4), 369–388 (Dec 2008)
15. Ji, Y., Massanari, R.M., Ager, J., Yen, J., Miller, R.E., Ying, H.: A fuzzy logic-based computational recognition-primed decision model. Inf. Sci. 177(20), 4338–4353 (Oct 2007)
16. Johnson, W.L., Rickel, J.W., Lester, J.C.: Animated pedagogical agents: Face-to-face interaction in interactive learning environments. INTERNATIONAL JOURNAL OF ARTIFICIAL INTELLIGENCE IN EDUCATION 11, 47–78 (2000)
17. Kao, E., Chang, P., Chien, Y.H., Soo, V.W.: Using ontology to establish social context and support social reasoning. In: Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., Rist, T. (eds.) Intelligent Virtual Agents. LNCS, vol. 3661, pp. 344–357. Springer Berlin / Heidelberg (2005)
18. Li, W., Allbeck, J.M.: Populations with purpose. In: Proc. of the Fourth International Conference on Motion in Games (MIG 2011). pp. 132–143. Springer (November 2011)
19. Luo, L., Zhou, S., Cai, W., Low, M.Y.H., Tian, F., Wang, Y., Xiao, X., Chen, D.: Agent-based human behavior modeling for crowd simulation. Computer Animation and Virtual Worlds 19(3-4), 271–281 (2008)
20. McGinnies, E.: Perspectives on Social Behavior. Gardner Press, Inc. (1994)
21. Mitchell, T.M., Keller, R.M., Kedar-Cabelli, S.T.: Explanation-based generalization: A unifying view. Machine Learning 1, 47–80 (1986)
22. Orkin, J., Roy, D.: Automatic learning and generation of social behavior from collective human gameplay. In: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems. pp. 385–392. AAMAS ’09 (2009)
23. Paris, S., Donikian, S.: Activity-driven populace: a cognitive approach to crowd simulation. IEEE Comput. Graph. Appl. 29(4), 34–43 (Jul 2009)
24. Pelechano, N., O’Brien, K., Silverman, B., Badler, N.I.: Crowd simulation incorporating agent psychological models, roles and communication. In: First International Workshop on Crowd Simulation. pp. 21–30 (2005)

25. Pinker, S.: The Language Instinct. HarperCollins (1995)
26. Rao, A.S., Georgeff, M.P.: Modeling rational agents within a bdi-architecture (1991)
27. Russell, S., Norvig, P.: Articial Intelligence: A Modern Approach (3rd edition). Prentice-Hall (2009)
28. Shao, W., Terzopoulos, D.: Autonomous pedestrians. In: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. pp. 19–28 (2005)
29. Sklar, E., Richards, D.: The use of agents in human learning systems. In: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems. pp. 767–774. AAMAS '06 (2006)
30. Stark, R.: Sociology. Thomson Wadsworth Publishing (2006)
31. Tom, M.: Machine Learning. McGraw Hill (1997)
32. Wang, Z., Lee, J., Marsella, S.: Towards more comprehensive listening behavior: Beyond the bobble head. In: Vilhjálmsson, H., Kopp, S., Marsella, S., Thórisson, K. (eds.) Intelligent Virtual Agents. Lecture Notes in Computer Science, Springer Berlin / Heidelberg
33. Yu, Q., Terzopoulos, D.: A decision network framework for the behavioral animation of virtual humans. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA). pp. 119–128 (2007)