

AI Framework for Decision Modeling in Behavioral Animation of Virtual Avatars

A. Iglesias¹ and F. Luengo²

¹ Department of Applied Mathematics and Computational Sciences, University of Cantabria, Avda. de los Castros, s/n, 39005, Santander, Spain

² Department of Computer Science, University of Zulia, Post Office Box #527, Maracaibo, Venezuela

iglesias@unican.es, fluengo@cantv.net

Abstract. One of the major current issues in Artificial Life is the *decision modeling* problem (also known as *goal selection* or *action selection*). Recently, some Artificial Intelligence (AI) techniques have been proposed to tackle this problem. This paper introduces a new based-on-Artificial-Intelligence framework for decision modeling. The framework is applied to generate realistic animations of virtual avatars evolving autonomously within a 3D environment and being able to follow intelligent behavioral patterns from the point of view of a human observer. Two examples of its application to different scenarios are also briefly reported.

1 Introduction

The realistic simulation and animation of the behavior of virtual avatars emulating human beings (also known as *Artificial Life*) has attracted much attention during the last few years [2,5,6,7,8,9,10,11,12,13]. A major goal in behavioral animation is the construction of an “intelligent” system able to integrate the different techniques required for the realistic simulation of the behavior of virtual humans. The challenge is to provide the virtual avatars with a *high degree of autonomy*, so that they can evolve freely, with a minimal input from the animator. In addition, this animation is expected to be *realistic*; in other words, the virtual avatars must behave according to reality from the point of view of a human observer.

Recently, some Artificial Intelligence (AI) techniques have been proposed to tackle this problem [1,3,4,8]. This paper introduces a new based-on-Artificial-Intelligence framework for decision modeling. In particular, we apply several AI techniques (such as neural networks, expert systems, genetic algorithms, K-means) in order to create a sophisticated behavioral system that allows the avatars to take intelligent decisions by themselves. The framework is applied to generate realistic animations of virtual avatars evolving autonomously within a 3D environment and being able to follow intelligent behavioral patterns from the point of view of a human observer. Two examples of the application of this framework to different scenarios are briefly reported.

The structure of this paper is as follows: the main components of our behavioral system are described in detail in Section 2. Section 3 discusses the performance of this approach by means of two simple yet illustrative examples. Conclusions and future lines in Section 4 close the paper.

2 Behavioral System

In this section the main components of our behavioral system are described.

2.1 Environment Recognition

At the first step, a virtual world is generated and the virtual avatars are placed within. In the examples described in this paper, we have chosen a virtual park and a shopping center, carefully chosen environments that exhibit lots of potential objects-avatars interactions. In order to interact with the 3D world, each virtual avatar is equipped with a perception subsystem that includes a set of individual sensors to analyze the environment and capture relevant information. This analysis includes the determination of distances and positions of the different objects of the scene, so that the agent can move in this environment, avoid obstacles, identify other virtual avatars and take decisions accordingly. Further, each avatar has a predefined vision range (given by a distance threshold value determined by the user), and hence, objects far away from the avatar are considered to be visible only if the distance from the avatar to the object is less than such threshold value; otherwise, the object becomes invisible.

All this information is subsequently sent to an analyzer subsystem, where it is processed by using a representation scheme based on genetic algorithms. This scheme has proved to be extremely useful for pattern recognition and identification. Given a pair of elements A and B and a sequence j , there is a distance function that determines how near these elements are. It is defined as $dist(j, A, B) = \frac{1}{k} \sum_{i=1}^k |A_i^j - B_i^j|$, where A_i^j denotes the i th gene at sequence j for the chromosome A , and k denotes the number of genes of such a sequence. Note that we can think of sequences in terms of levels in a tree. The sequence j is simply the level j down the tree at which it appears, with the top of the tree as sequence 1. A and B are *similar at sequence* (or *at level*) j if $dist(j, A, B) = 0$. Note that this hierarchical structure implies that an arbitrary object is nearer to that minimizing the distance at earlier sequences. This simple expression provides a quite accurate procedure to classify objects at a glance, by simply comparing them sequentially at each depth level.

2.2 Knowledge Acquisition

Once new information is attained and processed by the analyzer, it is sent to the knowledge motor. This knowledge motor is actually the “brain” of our system. Its main components are depicted in Figure 1(left). Firstly, the current information

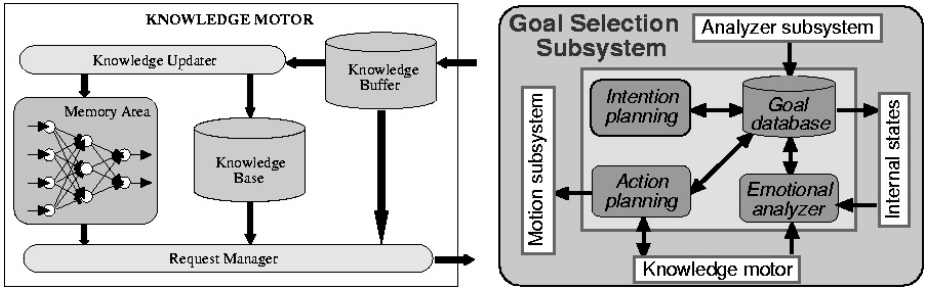


Fig. 1. (left) Knowledge motor scheme; (right) goal selection subsystem scheme

is temporarily stored into the *knowledge buffer*, until new information is attained. At that time, previous information is sent to the *knowledge updater* (KU), the new one being stored into this knowledge buffer and so on. This KU updates both the memory area and the knowledge base.

The *memory area* is a neural network applied to learn from data (in our problem, the information received from the environment through the perception subsystem). In this paper we consider the unsupervised learning, and hence we use an autoassociative scheme, since the inputs themselves are used as targets. To update the memory area, we employ the K-means least-squares partitioning algorithm for competitive networks, which are formed by an input and an output layer, connected by feed forward connections. Each input pattern represents a point in the configuration space (the space of inputs) where we want to obtain classes. This type of architecture is usually trained with a *winner takes all* algorithm, so that only those weights associated with the output neuron with largest value (the winner) are updated. The basic algorithm consists of two main steps: (1) compute cluster centroids and use them as new cluster seeds and (2) assign each chromosome to the nearest centroid. The basic idea behind this formulation is to overcome the limitation of having more data than neurons by allowing each neuron to store more than one data at the same time.

The *knowledge base* is actually a based-on-rules expert system, containing both concrete knowledge (facts) and abstract knowledge (inference rules). Facts include complex relationships among the different elements (relative positions, etc.) and personal information about the avatars (personal data, schedule, hobbies or habits), i.e. what we call *avatar's characteristic patterns*. Additional subsystems for tasks like learning, coherence control, action execution and others have also been incorporated. This deterministic expert system is subsequently modified by means of probabilistic rules, for which new data are used in order to update the probability of a particular event. Thus, the neuron does not exhibit a deterministic output but a probabilistic one: what is actually computed is the probability of a neuron to store a particular data at a particular time. This probability is continuously updated in order to adapt our recalls to the most recent data. This leads to the concept of *reinforcement*, based on the fact that the repetition of a particular event over time increases the probability to recall it.

Of course, some particular data are associated with high-relevance events whose influence does not decrease over time. A learning rate parameter introduced in our scheme is intended to play this role.

Finally, the *request manager* is the component that, on the basis of the information received from the previous modules, provides the information requested by the goal selection subsystem described in next section.

2.3 Decision Modeling

A central issue in behavioral animation is the adequate choice of appropriate mechanisms for *decision modeling*. Those mechanisms will take a decision about which is the next action to be carried out from a set of feasible actions. The fundamental task of any decision modeling module is to determine a based-on-priority sorted list of goals to be performed by the virtual agent. The goal's priority is calculated as a combination of different avatar's internal states (given by mathematical functions not described in this paper because of limitations of space) and external factors (which will determine the goal's feasibility).

Figure 1(right) shows the architecture of our goal selection subsystem, comprised of three modules and a *goal database*. The database stores a list of arrays (associated with each of the available goals at each time) comprised of: the goal ID, its feasibility rate (determined by the analyzer subsystem), the priority of such a goal, the wish rate (determined by the emotional analyzer), the time at which the goal is selected and its success rate.

The *emotional analyzer (EA)* is the module responsible to update the wish rate of a goal (regardless its feasibility). Such a rate takes values on the interval $[0, 100]$ according to some mathematical functions (not described here) that simulate human reactions in a very realistic way (as shown in Section 3).

The *intention planning (IP)* module determines the priority of each goal. To this aim, it uses information such as the factibility and wish rate. From this point of view, it is rather similar to the "intention generator" of [13] except by the fact that decision for that system is exclusively based on rules. This module also comprises a buffer to store temporarily those goals interrupted for a while, so that the agent exhibits a certain "persistence of goals". This feature is specially valuable to prevent avatars from the oscillatory behavior appearing when the current goal changes continuously.

The last module is the *action planning (AP)*, a based-on-rules expert system that gets information from the environment (via the knowledge motor), determines the sequence of actions to be carried out in order to achieve a particular goal and updates the goal's status accordingly.

2.4 Action Planning and Execution

Once the goals and priorities are defined, this information is sent to the motion subsystem to be transformed into motion routines (just as the orders of our brain are sent to our muscles) and then animated in the virtual world. Currently, we

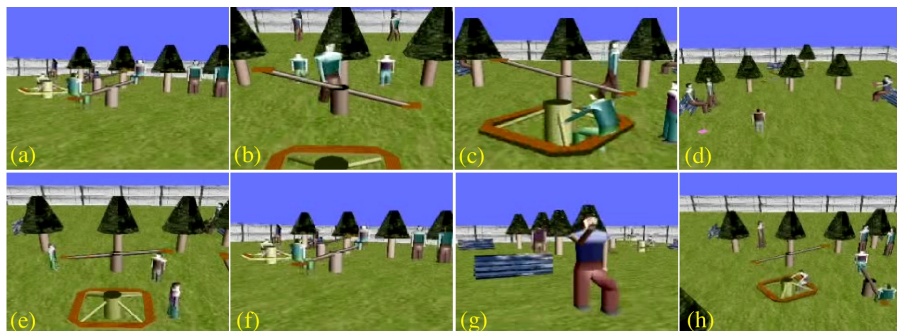


Fig. 2. Example 1: screenshots of the virtual park environment

have implemented routines for path planning and obstacle avoidance. In particular, we have employed a modification of the A* path finding algorithm, based on the idea to prevent path recalculation until a new obstacle is reached. This simple procedure has yielded substantial savings in time in all our experiments. In addition, sophisticated collision avoidance algorithms have been incorporated into this system (see the examples described in Section 3).

3 Two Illustrative Examples

In this section, two illustrative examples are used to show the good performance of our approach. The examples are available from Internet at the URLs: [http://personales.unican.es/iglesias/CGGM2007/sample \$x\$.mov](http://personales.unican.es/iglesias/CGGM2007/samplex.mov) ($x = 1, 2$). Figure 2 shows some screenshots from the first movie. In picture (a) a woman and her two children go into the park. The younger kid runs following some birds. After failing to capture them, he gets bored and joins his brother. Then, the group moves towards the wheel avoiding the trees and the seesaw (b). Simultaneously, other people (the husband and a girl) enter into the park. In (c) a kid is playing with the wheel while his brother gets frustrated after expecting to play with the seesaw (in fact, he was waiting for his brother besides the seesaw). After a while, he decides to join his brother and play with the wheel anyway. Once her children are safely playing, the woman relaxes and goes to meet her husband, who is seated on a bench (d). The girl is seated in front of them, reading a newspaper. Two more people go into the park: a man and a kid. The kid goes directly towards the playground, while the man sees the girl, becomes attracted by her and decides to sit down on the same bench, looking for a chat. As she does not want to chat with him, she stands up and leaves. The new kid goes to play with the wheel while the two brothers decide to play with the seesaw. The playground has two seesaws, so each brother goes towards the nearest one (e). Suddenly, they realize they must use the same one, so a brother changes his trajectory and moves towards the other seesaw. The mother is coming back in order to take after her children. Her husband also comes behind her and they

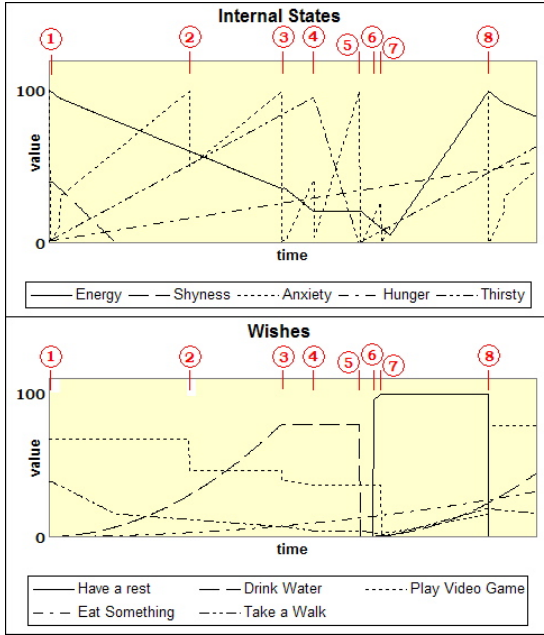


Fig. 3. Temporal evolution of the internal states (top) and available goals' wishes (bottom) for the second example in this paper

start to chat again (f). The man on a bench is now alone and getting upset so he decides to take a walk and look for the girl again. Simultaneously, she starts to make physical exercises (g). When the man realizes she's busy and hence will not likely pay attention on him, he changes his plans and walks towards the couple, who are still chatting (g). The man realizes they are not interested to chat with him either, so he finally leaves the park.

It is interesting to point out that the movie includes a number of remarkable motion and behavioral features. For instance, pictures (a)-(b)-(g) illustrate several of our motion algorithms: persecution, obstacle avoidance, path finding, interaction with objects (wheel, seesaw, bench) and other avatars, etc. People in the movie exhibit a remarkable ability to capture information from the environment and change their trajectories in real time. On the other hand, they also exhibit a human-like ability to realize about what is going on about others and change their plans accordingly. Each virtual avatar has previous knowledge on neither the environment nor other avatars, as it might happen in real life when people enter for the first time into a new place or know new people.

The second scene consists of a shopping center at which the virtual avatars can perform a number of different actions, such as eat, drink, play videogames, sit down to rest and, of course, do shopping. We consider four virtual avatars: three kids and a woman. The pictures in Figure 3 are labelled with eight numbers indicating the different simulation's milestones (the corresponding animation screenshots for those time units are displayed in Figure 4): (1) at the initial

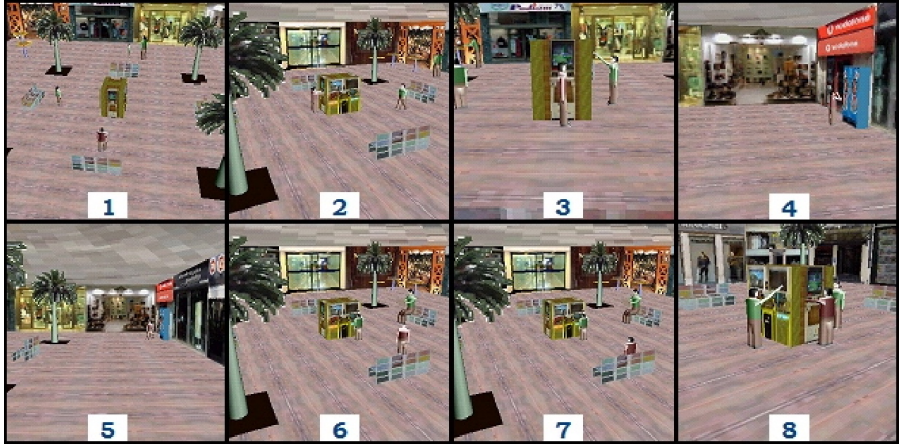


Fig. 4. Example 2: screenshots of the shopping center environment

step, the three kids go to play with the videogame machines, while the woman moves towards the eating area (indicated by the tables in the scene). Note that the internal state with the highest value for the avatar analyzed in this work is the energy, so the avatar is going to perform some kind of dynamic activity, such as to play; (2) the kid keeps playing (and their energy level going down) until his/her satisfaction reaches the maximum value. At that time, the anxiety increases, and avatar's wish turns into performing a different activity. However, the goal **play videogame** has still the highest wish rate, so it will be in progress for a while; (3) at this simulation step, the anxiety reaches a local maximum again, meaning that the kid is getting bored about playing videogames. Simultaneously, the goal with the highest value is **drink water**, so the kid stops playing and looks for a drink machine; (4) at this time, the kid gets the drink machine, buys a can and drinks. Consequently, the internal state function **thirsty** decreases as the agent drinks until the status of this goal becomes *goal attained*; (5) Once this goal is satisfied, the goal **play videogames** is the new current goal. So, the kid comes back towards the videogame machines; (6) however, the energy level is very low, so the goal **play videogames** is interrupted, and the kid looks for a bench to sit down and have a rest; (7) once seated, the energy level turns up and the goal **have a rest** does not apply anymore; (8) since the previous goal **play videogames** is still in progress, the agent comes back and plays again.

Figure 3 shows the temporal evolution of the internal states (top) and the goals' wishes (bottom) for one of the kids. Similar graphics can be obtained for the other avatars (they are not included here because of limitations of space). The picture on the top displays the temporal evolution of the five internal state functions (valued onto the interval $[0, 100]$) considered in this example, namely, **energy**, **shyness**, **anxiety**, **hunger** and **thirsty**. On the bottom, the wish rate (also valued onto the interval $[0, 100]$) of the feasible goals (**have a rest**, **eat something**, **drink water**, **take a walk** and **play videogame**) is depicted.

4 Conclusions and Future Work

The core of this paper is the realistic simulation of the human behavior of virtual avatars living in a virtual 3D world. To this purpose, the paper introduces a behavioral system that uses several Artificial Intelligence techniques so that the avatars can behave in an intelligent and autonomous way. Future lines of research include the determination of new functions and parameters to reproduce human actions and decisions and the improvement of both the interaction with users and the quality of graphics. Financial support from the Spanish Ministry of Education and Science (Project Ref. #TIN2006-13615) is acknowledged.

References

1. Funge, J., Tu, X., Terzopoulos, D.: Cognitive modeling: knowledge, reasoning and planning for intelligent characters, SIGGRAPH'99, (1999) 29-38
2. Geiger, C., Latzel, M.: Prototyping of complex plan based behavior for 3D actors, Fourth Int. Conf. on Autonomous Agents, ACM Press, NY (2000) 451-458
3. Granieri, J.P., Becket, W., Reich, B.D., Crabtree, J., Badler, N.I.: Behavioral control for real-time simulated human agents, Symposium on Interactive 3D Graphics, ACM, New York (1995) 173-180
4. Grzeszczuk, R., Terzopoulos, D., Hinton, G.: NeuroAnimator: fast neural network emulation and control of physics-based models. SIGGRAPH'98 (1998) 9-20
5. Iglesias A., Luengo, F.: New goal selection scheme for behavioral animation of intelligent virtual agents. IEICE Trans. on Inf. and Systems, **E88-D**(5) (2005) 865-871
6. Luengo, F., Iglesias A.: A new architecture for simulating the behavior of virtual agents. Lectures Notes in Computer Science, **2657** (2003) 935-944
7. Luengo, F., Iglesias A.: Framework for simulating the human behavior for intelligent virtual agents. Lectures Notes in Computer Science, **3039** (2004) Part I: Framework architecture. 229-236; Part II: Behavioral system 237-244
8. Monzani, J.S., Caicedo, A., Thalmann, D.: Integrating behavioral animation techniques. EUROGRAPHICS'2001, Computer Graphics Forum **20**(3) (2001) 309-318
9. Raupp, S., Thalmann, D.: Hierarchical model for real time simulation of virtual human crowds. IEEE Trans. Visual. and Computer Graphics. **7**(2) (2001) 152-164
10. Sanchez, S., Balet, O., Luga, H., Dutheu, Y.: Autonomous virtual actors. Lectures Notes in Computer Science **3015** (2004) 68-78
11. de Sevin, E., Thalmann, D.: The complexity of testing a motivational model of action selection for virtual humans, Proceedings of Computer Graphics International, IEEE CS Press, Los Alamitos, CA (2004) 540-543
12. Thalmann, D., Monzani, J.S.: Behavioural animation of virtual humans: what kind of law and rules? Proc. Computer Animation 2002, IEEE CS Press (2002)154-163
13. Tu, X., Terzopoulos, D.: Artificial fishes: physics, locomotion, perception, behavior. Proceedings of ACM SIGGRAPH'94 (1994) 43-50