In Proceedings of the European Conference on Artificial Life, ECAL'99 ,
Lausanne, Switzerland, 1999, 345-349

Level of Autonomy for Virtual Human Agents

Soraia Raupp Musse, Marcelo Kallmann and Daniel Thalmann

Computer Graphics Lab. Swiss Federal Institute of Technology
EPFL, DI-LIG, CH 1015 Lausanne, Switzerland
{soraia,kallmann,thalmann }@ lig.di.epfl.ch

Abstract. This paper proposes a paradigm for specification of virtual human agents' level of autonomy. The idea we present in this paper aims at optimising the required complexity of agents in order to perform realistic simulations. The paradigm is based on the distribution of the required virtual human agent "intelligence" to other simulation entities like groups of agents, and objects.

1 Introduction

Several methods have been introduced to model learning processes, perceptions, actions, behaviours, etc, in order to build more intelligent and autonomous virtual agents. Our goal in this paper is to propose a paradigm to define virtual agents endowed with different degrees of behavioural autonomy.

First of all, we present some useful concepts assumed in this work. A *virtual human agent* (here after just referred to as an agent) is a humanoid whose behaviours are inspired by those of humans' [15]. The term *group* will be used to refer to a group of agents, and the term *object* for an interactive object of the environment. Agents, groups, and objects constitute the *entities of the simulation*. A high-level *behavioural autonomy* concerns the ability to simulate complex behaviours. In this paper, we consider that the ability of agents for autonomously acting can be included in the agents (agents-based application), groups (groups-based application) or in the objects (objects-based application). Among others, interactivity, complex behaviours, intelligent abilities and frame rate of execution are directly related to the *level of autonomy* (LOA). Table 1 presents this relation using three kinds of behavioural control: *guided* represents the lower level of autonomy where the behaviours have to be provided by an external process. Yet, in Table 1, *programmed* control implies to use a notation (language) to define possible behaviours. The *autonomous* behaviour concerns the capability of acting independently exhibiting control over their internal state [28].

Table 1. Characteristics of different levels of autonomy (LOA).

BEHAVIOUR CONTROL	GUIDED	PROGRAMMED	AUTONOMOUS
Level of Autonomy	Low	Medium	High
Level of Intelligence	Low	Medium	High
Execution frame-rate	Low	Medium	High
Complexity of behaviours	Low	Variable	High
Level of Interaction	High	Variable	Variable

2 Related Work

Several works have discussed the various manners to simulate and interact with virtual agents. Zeltzer [29] presents a classification of levels of interaction and abstraction required in different applications. Thalmann [4] proposes a new classification of synthetic actors according to the method of controlling motion. Reynolds [22] presented the aggregated motions modelling. In recent work, a crowd model has been introduced using different abstractions of behaviours, like the term *guided crowd* [17]. Considering agent-object interaction tasks, some semantic information has been included within the object description. In particular, the *object specific reasoning* [11] creates a relational table to inform object purpose, and *smart objects* were introduced [10] containing interaction information.

3 LOA Related to Individuals

Several works agree with the concept of autonomous or "intelligent" agent requirements: autonomous behaviour, action, perception, memory, reasoning, learning, self-controlled, etc [15], [18], [22]. Yet, a lot of methods have been developed in order to model autonomous agents: L-systems [18], vision systems [21]; rule-based systems [22]; learning methods [25], etc. Yet, guided or programmed agents can also be useful depending on the application. Table 2 exemplifies the three kinds of agent autonomy using two different agent tasks.

Table 2. LOA present in different agent-oriented tasks.

LOA/TASK	AGENT GOES TO A SPECIFIC LOCATION	AGENT APPLIES A SPECIFIC ACTION
Guided	Agent needs to receive during the simulation a list of collision-free positions	Agent needs to receive information about the action to be applied
Programmed	Agent is programmed to follow a path while avoiding collision with other agents and programmed obstacles	Agent is programmed to manage where and how the action can occur
Autonomous	Agent is able to perceive information in the environment and decide a path to follow to reach the goal, using the environment perception or the memory (past experiences)	Agent can decide about an action to be applied. This action can be programmed, imitated or existent in the memory (past experiences)

4 LOA Related to Groups of Agents

In the case of crowd simulation, usually we intend to have lots of virtual human agents avoiding dealing with individual behaviours. Contrary to the last section, our goal here is to describe methods to provide intelligence focused in a common group entity that controls its individuals. We have called *groups-based application*, the crowd and group applications, where individual complexity is less required. In this case, the intelligence abstraction can be included in the groups providing more autonomy to the groups instead to the individuals.

Considering levels of autonomy (LOA), we have classified the crowd behaviours in three kinds: i) Guided crowds, which behaviours are defined explicitly by the users; ii) Programmed crowds, which behaviours are programmed in a script language; iii) Autonomous crowds, which behaviours are specified using rules or others complex methods. Table 5 exemplifies this classification of crowd autonomy using two different crowd tasks.

Table 3: LOA present in different group-oriented tasks

LOA/TASK	Group goes to a specific location	Group reacts to matched event
Guided	Group needs to receive during the simulation a list of positions "in-betweens" in order to reach the goal	Group needs to receive an information about the matched event and the reaction to be applied
Programmed	Group is programmed to follow a path avoiding collision with other agents and programmed obstacles.	Group can manage events and reactions, which are programmed.
Autonomous	Group is able to perceive information in the environment and decide a path to follow to reach the goal, using the environment perception or the memory (past experiences).	Group can perceive a matched event and decide about the reaction to be applied. This reaction can be also programmed or existent in the group memory.

5 LOA Related to Objects

Whenever the simulation needs to handle complex agent-object interactions, many difficult issues arise. Such difficulties are related to the fact that each object has its own movements, functionality and purposes. One can consider that agents' perceptions can solve some simple tasks, as for instance a single-hand automatic grasping of small objects. But this is no more possible for interactions with objects that have an intricate proper functionality, as the lift example. In fact, each time more information related to the object is given, its level of autonomy (LOA) is increased. Table 6 illustrates how an agent must proceed according to the different LOAs for three different interactive objects of the environment.

Table 4: LOA present in different objects-oriented tasks

LOA/OBJECT	Door	Sign	Lift
Guided	The agent have to move its arm to a attainable and meaning location of the door, and control its movement until open it.	The agent recognises that the sign has an arrow and recognises the showed direction.	The agent recognises where is the call button, how and when the door opens, how and where to enter inside the lift, when and how to go out, etc.
Programmed	The agent has to move its arm to the right place but the door opens by itself.	The agent recognises the sign, but the direction is given with no recognition.	The agent accesses the current lift state and decides only its moves accordingly.
Autonomous	The door takes control of the agent telling exactly the place to put its hand and the complete movement of the door	The sign gives a new direction to go for each agent that passes nearby.	The lift takes control of the movements of the agent and gives him a complete plan, based on primitive actions, to perform the interaction.

6 The Proposed Paradigm

As presented in the last sections we considers that the "intelligence" is not only included in the virtual human agents, but can be also included in groups and objects. Considering the abstraction levels: guided, programmed and autonomous behaviours, we present a schema that includes the entities group and object, as showed in Figure 1. We can so classify a simulation in terms of the autonomy distribution among its entities, i.e., a simulation (S_i), can be translated as a function of three components: agents, groups and objects: $\underline{S_i = f(\,LOA(Agents),\,LOA(Groups),\,LOA(Objects)\,)}$

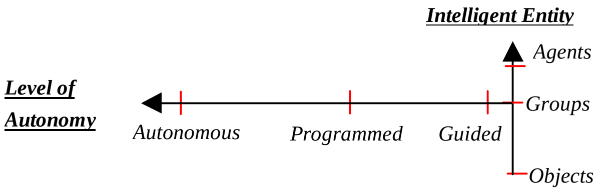

Figure 1: Level of autonomy vs. intelligent entity.

7 A Case Study

The chosen case study was designed to deal with different kinds of control. We consider the environment of a virtual city [6] containing some streets, a supermarket (S), a train station (TS), autonomous objects (direction signs to go to the TS) and other buildings. Let G be a group of virtual agents endowed with different LOAs, which can be guided, programmed or autonomous. The goal of group G is to go from the supermarket to the train station. We dealt with four kinds of simulations having different group controls, interacting or not with objects.

Facts: ~ Goal (G) = Go From S to TS	
Simulation 0 ~ G = Guided Group ~ Autonomous objects exist but G can not recognise them. ~ G does not know its goal ~ Initially, G receives a location to reach ~ G is able to walk to reach this location ~ G is not able to avoid collision with obstacles	Simulation 1 ~ G = Programmed group ~ Autonomous objects exist but G can not recognise them, because G is not programmed for that ~ G knows the programmed goal ~ Initially, G is able to translate a programmed goal (TS) in a path to be applied
Simulation 2 ~ G = Autonomous group. ~ Autonomous objects exist and G can recognise ~ G knows its goal ~ Initially, G is able to recognise the autonomous objects, go to a location near to it and interact with. The autonomous object is able to recognise where G wants to go and to give the correct direction ~G is able to follow object instructions	Simulation 4 ~G = Autonomous group ~ Autonomous objects do not exist ~ G knows its goal ~ G is provided with vision and environment knowledge ~ G can find a path to reach the goal by perceiving the environment (signs) by its own ~ G is able to perceive and avoid collision with obstacles

Figure 2: (left): the starting and goal points; (center) group G going to interact with autonomous object; (right): comparison data between the four simulations.

In Figure 2 (right), some parameters (except execution time) represent subjective data to be measured, then we decide to intuitively quantify it in four levels: 25 (Low), 50 (inferior medium), 75 (superior medium) and 100 (high).

8 Conclusions

We propose in this paper a paradigm to distribute the autonomy among the entities of the simulation. The idea we dealt here concerns the possibility to improve the frame rate of execution as well as to optimise the complexity required, by distributing some knowledge and autonomy to others entities of the simulation: groups and objects. This paradigm has been tested in the context of a Virtual City project [6] because we have to simulate several virtual human agents that can act in differently ways and apply different actions.

Acknowledgements

The research was sponsored by the Swiss National Research Foundation, the Federal Office of Education and Science in the framework of the European project eRENA [8], FUNDEPE, CAPES and CNPq - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Brazilian offices of Education and Science).

References

1. Cavazza, M., Earnshaw, R., Magnenat-Thalmann, N. and Thalmann, D. "Motion Control of Virtual Humans". IEEE CG&A, Vol. 18, No. 5, Sept/Oct 1998, pp.24-31.
2. Farenc, N; et al. "One Step Towards Virtual Human Management for Urban Environment Simulation". ECAI'98, Brighton, England. 1998.
3. Kallmann, M. and Thalmann, D. "Modeling Objects for Interaction Tasks", Proc. Eurographics Workshop on Animation and Simulation, 1998.
4. Levison, L. "Connecting Planning and Acting via Object-Specific reasoning", PhD thesis, Dept. of Computer & Information Science, University of Pennsylvania, 1996.
5. Meyer, J.A.; Guillot, A. "From SAB90 to SAB94: Four Years of Animat Research". In: Proceedings of Third International Conference on Simulation of Adaptive Behavior. Brighton, England, 1994.
6. Musse, S.R., Babski, C., Capin, T. and Thalmann, D. "Crowd Modelling in Collaborative Virtual Environments". ACM VRST /98, Taiwan
7. Noser, H. "A Behavioral Animation System based on L-Systems and Synthetic Sensors for Actors". PhD thesis. EPFL, Lausanne, Switzerland, 1996.
8. Renault, O., Magnenat-Thalmann, N. Thalmann, D. "A Vision-based Approach to Behavioral Animation", The Journal of Visualization and Computer Animation, Vol. 1, No. 1, pp. 18 – 21.
9. Reynolds, C. "Flocks, Herds and Schools: A Distributed Behavioral Model". Proc. SIGGRAPH '87, Computer Graphics, v.21, n.4, July, 1987.
10. Sutton, R.S. "Reinforcement Learning Architectures for Animats". In: Proceedings of First International Conference on Simulation of Adaptive Behavior. MTI Press, 1990.
11. Wooldridge, M. and Jennings, N. "Intelligent Agents: Theory and Practice". Knowledge Engineering Review, 10(2), June 1995. Cambridge University Press, 1995.
12. Zeltzer, D. "Task-level Graphical Simulation: Abstraction, Representation and Control". Making them Move: Mechanics, Control and Animation of Articulated Figures. Edited by N. Badler, B. Barsky and D. Zeltzer. Pp 3-33. 1991.

Filename:	ECALpub.doc
Directory:	Z:\mac_pc\Soraia\ECAL99
Template:	C:\Antje\temp\TestWV\sv-lncs.dot
Title:	Level of Autonomy for Virtual Human Agents
Subject:	
Author:	So&Mito
Keywords:	
Comments:	
Creation Date:	05/20/99 5:02
Change Number:	13
Last Saved On:	05/30/99 5:08
Last Saved By:	soraia
Total Editing Time:	211 Minutes
Last Printed On:	05/30/99 5:22
As of Last Complete Printing	
Number of Pages:	5
Number of Words:	2,025 (approx.)
Number of Characters:	11,546 (approx.)