Behavioral Programming of Autonomous Characters based on Probabilistic Automata and Personality

Luca Chittaro, Milena Serra
HCI Lab, Dept. of Math and Computer Science,
University of Udine,
Via delle Scienze 206, 33100 Udine, Italy
ph: +39 (0432) 558.450, fax: +39 (0432) 558.450
e-mail: {chittaro, serra}@dimi.uniud.it

This paper presents a system for realistic behavioral programming of virtual characters, based on personality and probabilistic automata. We describe personality by using the Five-Factor Model and achieve autonomy through a goal-oriented approach. Each character perceives the surrounding world, decides how to behave and acts on the environment according to its personality and to its goals. The chief idea explored by the proposed approach is that personality has a probabilistic influence on behavior selection instead of a deterministic one. Different behavior sequences available to achieve a goal are modeled using probabilistic automata and making probability dependent on character personality. This leads to non-repetitive behaviors, whose evolution is not foreseeable. The paper first motivates the approach in the context of cybertherapy. Then, it summarizes related work and illustrates in detail the proposed approach. Finally, it presents obtained results and discusses the main limitations of the implemented system.

Keywords: virtual characters, personality, behavioral autonomy, probabilistic automata

This is a preprint of an article published in the Journal of Computer Animation and Virtual Worlds,
Volume 15, Issue 3-4 (July 2004), p 319-326,

http://www.interscience.wiley.com/cgi-bin/jhome/106562739

Introduction

A very important challenge for the virtual reality community is to create Virtual Environments (VEs) inhabited by *realistic* characters. Realism should ideally be achieved in all layers of the modeling pyramid, i.e. geometric, kinematic, physical, behavioral and cognitive [1]. While there is a significant amount of work related to the first three levels, behavioral and cognitive modeling have become subjects of study in more recent times. The behavioral layer concerns abilities such as self-animation as reaction to environmental stimuli, while the cognitive layer concerns aspects such as characters knowledge, knowledge acquisition and planning abilities.

This paper is mainly concerned with realistic behavioral programming and the proposed approach has been developed in the context of a cybertherapy project. The word *cybertherapy* identifies psychological therapies that use virtual reality to create the situations needed to elicit particular emotions or states in a patient. One of the most significant successes in this field has been reached in the treatment of phobias: patients are exposed to objects and situations that cause their phobia, but this exposure takes place in a VE instead of the real world (Virtual Reality Exposure Therapy, VRET) [2, 3]. An important factor for the effectiveness of cybertherapy is the sense of *presence* felt by patients, that is highly correlated with VE realism [4]. As far as social phobias are concerned, VEs need to be populated by highly realistic characters, but realism is complex to achieve. Two of the main ideas that contribute to characters realism are *behavioral characterization*, i.e. differentiating characters behavior, and *autonomy*, i.e. the ability for a character to act on and react to the environment.

Characters should ideally be able to reproduce all the different behavioral traits of human beings. For example, when we observe a group of people, we do not expect them to move and behave all alike, event if they are in the same social context. In human beings, behavioral differences depend on several psychological factors, such as personality, mood and habits. Endowing characters with a psychological characterization determines a differentiation of their behavior. The problem of characters autonomy in cybertherapy has been pointed out by Gaggioli et al. [5], who discuss the use of virtual characters in clinical psychology. For example, if a character remains frozen, its behavior will not be consistent with that of real people. An autonomous character should instead behave according to what happens around it, as human beings do.

This paper is devoted to describe our solutions to the two mentioned problems, i.e. discriminating characters behavior through a psychological characterization and endowing characters with behavioral autonomy. It proposes a goal-oriented approach, based on *personality definition* and on *probabilistic automata*, to obtain a probabilistic influence of personality on behavior selection. The paper is organized as follows: first, we discuss related work in behavioral characterization and behavioral autonomy; then, we describe our approach and the underlying architecture; finally, the obtained results are evaluated and discussed .

Related work

Even if behavioral characterization and autonomy have become subjects of research relatively recently, there is a considerable amount of carried out studies. This section presents some representative examples of proposed approaches for the two problems.

Behavioral characterization

Different solutions have been proposed to the problem of improving believability of virtual characters and many of them concern psychological aspects, such as:

- *personality*, i.e. the total sum of all the behavioral and mental characteristics by means of which an individual is recognized as being unique;
- *emotional state*, i.e. the combination of all emotions felt in an instant, where an emotion is a strong feeling accompanied by a physical alteration (e.g. turning pale, blushing…);
- *mood*, i.e. a persistent emotional state.

Representing these aspects and mapping them into visible effects that the user notices are the first problems encountered in endowing a character with a psychological characterization. In particular, a way to describe personality is the Five-Factor Model (FFM) [6], a well-known psychological model that summarizes personality traits in five continuous dimensions (*openness*, *conscientiousness*, *extraversion*, *agreeableness* and *neuroticism*), whose values range from 0 to 100.

The *Character Markup Language* (*CML*) [7] is an example of a language for scripting the animation of virtual characters taking into account personality and emotions. In CML, a character can be defined as belonging only to one of the two extremes of the FFM dimensions (e.g. open-minded or close-minded, extravert or introvert…), while emotions are separately defined. This way, the effects of emotions on character voice, facial expression and behavior are algorithmically generated. The FFM is used also in the *EMOTE* system [8], where personality influences both the character perception of the environment and the way actions are performed, but not the behavioral choices. André [9] proposes to use both personality and emotions to select a behavior and to convey the internal state of a character through posture, speech and facial expression. Personality is related to emotions in such a way that personality is a variable that determines the intensity of emotions. Personality is defined using the FFM, while emotions are those provided by the OCC-Model [10]. Bécheiraz and Thalmann [11] created a system where an emotion is generated as a consequence of a perception. The emotion produces immediately a reaction, that can be a body response, a facial expression or a behavior selection. No personality definition is taken into account. A system comprising all the above mentioned psychological aspects (i.e. personality, emotional state and mood) has been proposed by Egges, Kshirsagar and Thalmann [12], who used it to create a conversational agent whose responses and facial expressions depend on all the three aspects. In the Improv system [13], information about the character can influence probability of choice of *scripts*, i.e. simple actions or sequences of actions/scripts. Character information is made of several variables that are freely created by the animator, thus the system lacks an homogeneous, explicit definition of personality and emotional state.

Behavioral autonomy

A character can be said to be *autonomous* when it is able to perceive objects and creatures in the VE and to behave accordingly to these perceptions [14]. Various approaches have been proposed to create autonomous characters. For example, in the mVITAL system [15], every character is provided with a small KBS (Knowledge-Based System) that defines what it knows (*initial beliefs*), what it can infer from its knowledge (*reasoning abilities*) and what it wants to do (*goals*). Such method is very flexible, but defining the knowledge base is a complex and time-consuming task, requiring significant expertise. A reasoning system is used also by Noser and Thalmann [16], who propose a system where motion decisions are based on perception, visual memory and reasoning abilities.

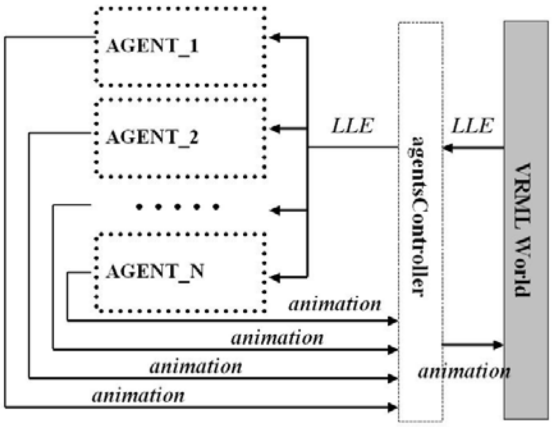

Figure 1: High-level architecture

In the Greta Project [17], the decisional structure of the character is represented by a set of Dynamic Belief Networks (DBN), each one modeling the probability for an emotion to arise. The emotional state then determines the next behavior to be chosen. Bécheiraz and Thalmann [11] use Finite-State Machines (FSMs) to describe and compose possible behaviors, whose selection is determined not only by the emotional state, but also by the perceptual state. FSMs are used also by Brooks [18] to create multiple layers for a robot architecture. Each layer is an isolated computational unit that implements the whole cognitive process, i.e. the *Sense-Decide-Act* (*SDA*) cycle: it is able to perceive what happens in the surrounding world, to respond to it by selecting a proper behavior and to act on the environment. A FSM is activated through sensor nodes and algorithmically produces an output through actuator nodes.

Proposed approach

In our approach, we define the personality of characters using the FFM [6]. A personality is defined assigning values between 0 and 100 to each of the five dimensions considered by the model. The internal processes of characters follow the SDA cycle, i.e. characters are able to perceive what happens around them, to decide a proper reaction and to perform the related actions in the VE. We achieve behavioral differentiation by having personality influence each of those three steps. Therefore, characters with different personality traits may perceive the world differently, may select different behaviors and may perform the same action in different ways.

Our approach to behavioral autonomy is goal-oriented: every decision taken by a character aims to achieve one or more goals, that are provided by the animator when defining the character. The behavior sequences that can be used to achieve a goal are modeled through probabilistic automata (*Probabilistic Finite-State Machines*, *PFSMs*), where probabilities are influenced by character personality, as we will see in the following. While in the previously described approaches the psychological characterization had a deterministic influence on characters behavior, in our approach this influence is *probabilistic*. The Improv approach [13] is able to introduce probabilistic behaviors, but it does not use an homogeneous and explicit personality definition and is not based on PFSMs.

Probabilistic personality influence implies that, as it happens with human beings, one cannot fully predict how a character will react to a stimulus: its personality might lead it to act most of the times in a certain way, but other reactions are possible. As a consequence, the same situation can evolve in different ways.

System architecture

Our system has been implemented in Java and relies on VRML for visualization purposes. The high-level architecture of the system is shown in Figure 1: each character in the VRML world is managed by a software agent; each of them is able to perceive the environment and the other characters through a flow of *Low-Level Events* (*LLEs*). LLEs describe sampled changes in position and orientation of each character, of the user and of objects in the VE. Moreover, they detect when a character is touched by the user. Characters can interact with the environment and other characters, resulting in the production of further events, and they perceive the user as they perceive any other character. A centralized module, the *agentsController*, plays the role of an interface between agents and the VRML World: it sends to the agents the LLEs received from the VE and executes in the VE the animations chosen by the agents. Moreover, it implements a time-sharing mechanism, allowing agents to progress in parallel.

The internal organization of an agent into modules is illustrated in Figure 2. In the following, we describe in detail each module, also discussing how personality influences characters behavior in each step of the SDA cycle.

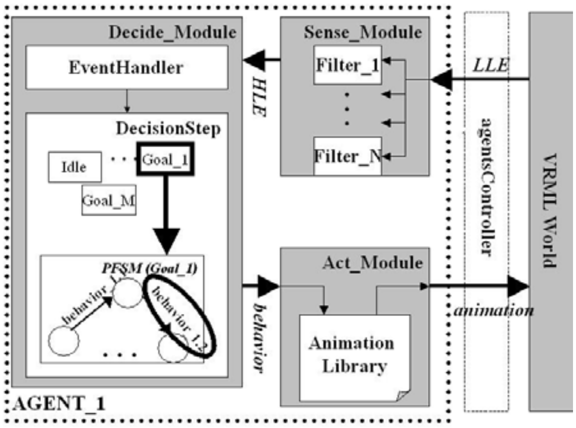

Figure 2: Internal architecture of a single agent

Sense

As far as *sensing* is concerned, we want each character to perceive the environment according to its position and personality. The *Sense_Module* (see Figure 2) contains a set of filters (one for the character field of view, one for its field of attention and one for the perception of being touched) that analyze environmental information, by processing LLEs coming from the VRML world, and determine if something new has to be noticed by the character. If this is the case, the *Sense_Module* generates an *High-Level Event* (*HLE*), i.e. an abstract description of interesting LLEs from the character point of view. As an example, the *FieldOfAttention* filter produces information on what gets the character attention. If the agent notices, by processing the LLEs, that the user's position has entered the agent field of attention, an HLE will indicate that the user has to be noticed by the character. The size of the field of attention depends on the character personality (openness factor): a very attentive character will have a wider field of attention than an absent-minded one. The set of filters can be extended and every new filter can use none, one or more personality factors to perform its task. This way, we achieve personality influence on the character perception.

Decide

People act to achieve one or more goals that arise from some external or internal stimuli. In the developed system, characters autonomy is accomplished through a goal-oriented approach: every character has a set of active goals and acts to achieve them. Each goal is activated or deactivated by a sub-module of the *Decide_Module* (i.e. the *EventHandler*) as a reaction to an HLE. For example, one can program an agent in such a way that when the *FieldOfAttention* filter communicates that the user has just got the character attention, a corresponding goal (e.g. "Interact with the user") is activated. A goal can be achieved in different ways, each one corresponding to a sequence of behaviors, where a *behavior* is an atomically executed action. In our approach, possible behavior sequences that achieve a goal are represented with a *PFSM*, where each *transition* (i.e. each edge in the automata) corresponds to a behavior. To each transition, we associate: i) a (possibly empty) list of preconditions that must be true to allow selecting the transition; ii) a *personality weights array* (*PWA*) containing the weight that each personality factor has on the probability of choosing the transition. Weights in the array are set to 0 by default; the animator has to assign a positive value only to those personality factors that have to affect the choice and she can refer to each personality factor both positively (e.g. extraversion) and negatively (e.g. introversion).

When an agent has to take a decision, it must first choose on which goal, among the activated ones, to focus; then, it has to choose a transition inside the related PFSM (see Figure 2). Both goal and transition are probabilistically chosen by the *DecisionStep* sub-module. A goal probability depends on: i) its *priority*, which is specified when the goal is defined; ii) its *recency of activation*, because a new goal should get the agent attention; iii) *conscientiousness* value in the character personality. Indeed, a very conscientious person will concentrate more on the goals she is trying to achieve, while a less conscientious one is more likely to be distracted by new goals. The probability of a transition depends on the associated PWA and on the character personality. For each transition, the probability of choice p is computed as follows: $p = \sum_{i=1}^{N} W[i] * P[i]$, where $W[i]$ is the i^{th} element of the N-dimensional PWA and $P[i]$ is the corresponding element in the personality definition.

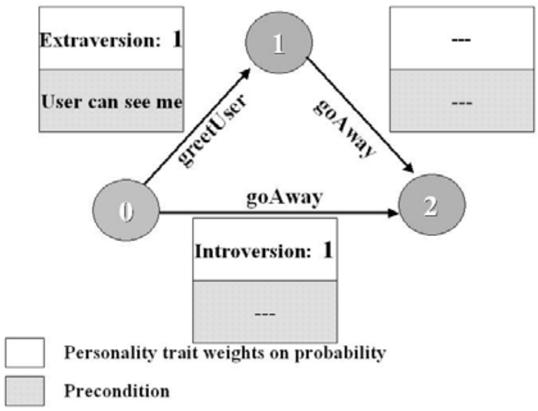

Figure 3: A simple PFSM related for the goal "React to Approaching User"

As an example, Figure 3 shows a very simple PFSM that can be related to a goal called "React to Approaching User". When this automata is first activated, it goes in state 0 and there are two possible behaviors: greeting the user or going away. To choose the greeting behavior, a precondition must be true, i.e. the user must be able to see the character. If the precondition is false, the character can only go away, otherwise the choice of behavior will be probabilistically determined. Extraversion/introversion is the only personality factor to which the animator has assigned a value and thus it is the only one that affects probability in this example. Let us assume that the personality of the character employing the PFSM of Figure 3 has an extraversion value of 80: this will be used as the normalized value 0.8 and the "greetUser" transition will have a probability of choice equal to 1×0.8 (1 is the extraversion weight in the PWA), while the "goAway" transition will have a probability of choice equal to 1×0.2 (introversion is computed as $1 - extraversion$ in the personality definition, 1 is the introversion weight in the PWA). Probabilities are always normalized before the choice takes place. By applying the

PFSM of Figure 3, an extraverted character is more likely to greet the user. However, there is a 20% probability that the opposite might happen and this represents the unpredictable nature of human behavior.

A character never remains frozen. Every time it completes a transition or every time a new event occurs, it takes a new decision and executes it; when no events occur and no goals are activated, there is always a goal to be chosen: the *Idle* goal. The purpose of this goal, which has no related PFSM, is to provide the character with a set of common actions, such as looking around, wandering or scratching its own head, that can be carried out when there is nothing more significant to do. The probability of choice of actions that are associated to the *Idle* goal depends on personality: each action has a PWA and the probability of choosing an action in the set is computed with the same formula described above. The *Idle* goal thus solves the problem of frozen characters described in the introduction.

Act

Personality of human beings influences the way they move and speak; e.g. an extravert person tends to make broader gestures compared to an introvert one [9]. In our system, each behavior is physically implemented by an animation (or a sequence of animations) stored in the *Animation Library* (see Figure 2) and we can associate alternative animations to the same behavior. The choice among alternatives is influenced by personality as follows: i) *ideal personality values* can be specified for each animation (the specification may involve none, one or more personality factors); ii) the *Act_Module* computes the Euclidean distance from the ideal personality values of animations to the corresponding elements in the character personality; iii) the minimum distance animation is chosen and executed in the VRML World. For example, there can be different types of animation for the walking behavior based on the extraversion value, but only the one with the extraversion value nearest to the character value will be chosen.

Two of the five factors influence additional aspects of how behaviors are carried out: neuroticism influences speed of animations, while extraversion influences the distance the character keeps from other characters.

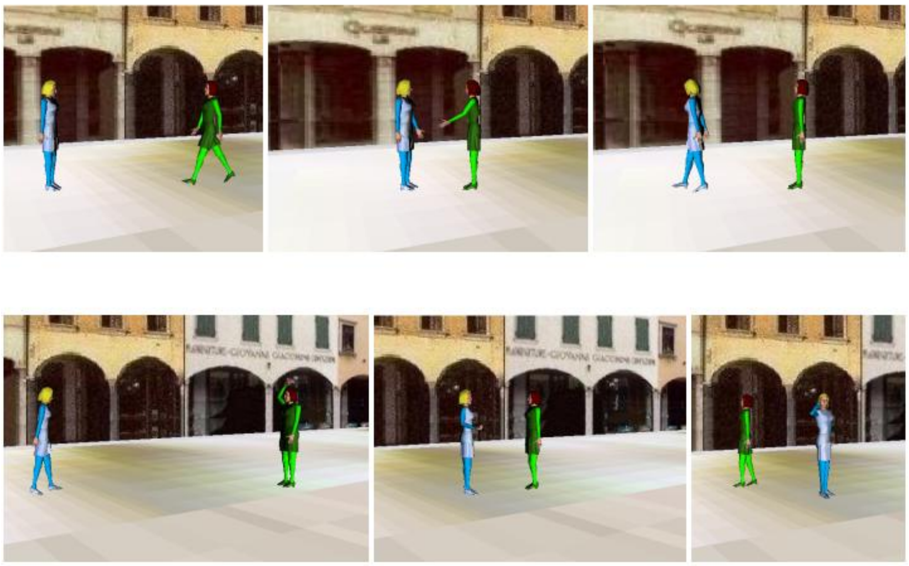

Figure 4: Two executions of the same scenario lead to different, but coherent results. UPPER SEQUENCE: the brown-haired character takes initiative, goes towards the other one and they have a conversation until the fair-haired character walks away. LOWER SEQUENCE: the fair-haired character takes initiative and goes towards the other one, but now interaction is limited to a brief greeting and the brown-haired character walks quickly away

Results

The probabilistic influence of personality factors, achieved through PFSMs, is effective in generating realistic behaviors. The generation of different behaviors, that are coherent with the general requirements provided by the animator (e.g. see the scenario described by Figure 4), makes it possible to use the same scenario several times. Moreover, avoiding repetitiveness increases the characters believability. These two results are very important in cybertherapy. Consider, for example, a scenario for the treatment of social phobia, where the patient is in a square full of characters and has to walk through the square. A single visit to the square cannot be sufficient and it can be necessary to repeat it several times. The therapist may decide to use the same scenario so that the patient get used to it. If the characters in the scenario do not vary their behavior, the scene will lose its believability and this can affect negatively the therapy.

Another positive aspect of our system is that the therapist can vary the typical behavior of a character by changing only the values of its personality factors (e.g. Figure 5 shows the effects of changing values of factors). With such a small effort, it is possible to obtain a set of characters that differ not only in what they do, but also in how they perceive events and in how they physically move. This way, characters behave more like real people, improving realism of the VE. Moreover the system is flexible, supporting the definition of deterministic behavior sequences, personality-influenced behavior sequences and even pseudo-random ones.

Informal evaluation

To informally evaluate the implemented system, we created four scenarios (some of which are suitable for the context of social phobia treatment), that contain some characters placed inside a reconstruction of a real city square. The informal evaluation we carried out aimed to test if (and which) differences among the characters behaviors are perceived by users. Moreover, we wanted to collect users' opinions to assess characters believability and realism. Thirteen university students, whose age varied from 20 to 25 years, were involved in the evaluation; 7 were males and 6 females; 6 of them had computer expertise and 6 had previous experience in navigating VEs. Following a think-aloud protocol, users were asked to try the described scenarios, explain what was happening, report similarities and differences they noticed among characters, remark what they liked or disliked about them and what they found highly or scarcely realistic.

The main results of this informal evaluation can be summarized as follows. First, 90% of users reported behavioral differences among characters with different personalities, even when the characters had the same goals (same PFSMs). It is interesting to note that users did comment on the differences by trying to give explanations in terms of emotions and/or moods (that were consistent with the differences in behavior). Second, users were more attracted by autonomous characters than by deterministically-programmed ones , that were also present in the scenario. For example, when in front of two characters, one deterministically-programmed and one probabilistically-programmed, all users followed shortly the first and then devoted all their attention to the other, whose behavior was not foreseeable.

Conclusions and future work

We proposed a goal-oriented approach to characters programming based on probabilistic automata. Characters are endowed with a personality that has a probabilistic influence on behavior selection. As a consequence, behavioral choices are not always predictable, consistently with real people. Moreover, personality influences also perception and actuation, generating characters that react in different ways to environmental changes and that perform the same type of action in different ways, similarly to what happens with human beings.

The major limitation of the proposed system is that characters have currently no long-term memory. They do not remember what goals they have achieved in the past, thus they can try to achieve them again, if a proper event activates them. This is desirable in some cases, e.g. the goal of avoiding a car. However, in some other cases, repetition reduces realism, e.g. if a character greets the user every time they meet but only a few seconds have passed since the last meeting. Therefore, our current priority is to extend the system in such a way that it becomes able to discriminate between the two types of goals. Moreover, we intend to investigate how to help animators in assigning PWAs to transitions in PFSMs to obtain believable behavior sequences. Currently, this is a trial-and-error process, that can be very time-consuming. A further improvement of our system may also include a module responsible for the emotional state of characters, so as to further extend their psychological model.

Figure 5: UPPER SEQUENCE: Interaction with an extravert and agreeable character: to call the character, the user clicks on it; the character comes promptly to the user; the character says something to the user. LOWER SEQUENCE: Interaction with a shy and disagreeable character: to call the character, the user clicks on it; the character ignores the user and walks away; the user goes near to the character and the latter shyly greets; the character walks immediately away

References

[1] J. Funge, X. Tu, and D. Terzopoulos. Cognitive Modeling: Knowledge, Reasoning and Planning for Intelligent Characters. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pages 29–38. ACM Press: New York, August 1999.

[2] B. K. Wiederhold and M. D. Wiederhold. The Use of Virtual Reality Therapy for The Treatment of Anxiety Disorders and Phobias. *Computer Graphics Journal of WSCG*, 33(2):25–27, May 1999.

[3] S. Bouchard and P. Renaud. Applications of Virtual Reality in Clinical Psychology: Illustrations With the Treatment of Anxiety Disorders. In *Proceedings of HAVE 2002 - IEEE International Workshop on Haptic Audio Visual Environments and Their Applications*, pages 7–11, November 2002.

[4] M. J. Schuemie. *Human Computer Interaction and Presence in Virtual Reality Exposure Therapy*. PhD thesis, Delft University of Technology, 2003. http://graphics.tudelft.nl/˜vrphobia/dissertation.pdf (last accessed on May 2004).

[5] A. Gaggioli, F. Mantovani, et al. Avatars in Clinical Psychology: A Framework for the Clinical Use of Virtual Humans. *CyberPsychology & Behavior*, 6(2):117–125, 2003.

[6] P. J. Howard and J. M. Howard. The Big Five Quickstart: An introduction to the Five-Factor Model of Personality for Human Resource Professionals, 1995. http://www.centacs.com/quickstart.htm (last accessed on May 2004).

[7] Y. Arafa and A. Mamdani. Scripting Embodied Agents Behavior with CML: Character Markup Language. In *International Conference on Intelligent User Interfaces*, pages 313–316. ACM Press: New York, January 2003.

[8] N. Badler, J. Allback, L. Zhao, and M. Byun. Representing and Parameterizing Agent Behaviors. In *Proceedings of Computer Animation*, pages 133–143. ACM Press: New York, January 2002.

[9] E. André. State of Art in Emotion and Personality. Presentation at Virtual Humans Workshop, March 2002. http://www.ict.usc.edu/˜vhumans/presentations/Andre-Emotions.ppt (last accessed on May 2004).

[10] A. Ortony, G. Clore, and A. Collins. *The Cognitive Structure of Emotion*. Cambridge University Press: New York, 1998.

[11] P. Bécheiraz and D. Thalmann. A Behavioral Animation System for Autonomous Actors personified by Emotions. In *Proceedings of First Workshop on Embodied Conversational Characters (WECC '98)*, pages 57–65, 1998.

[12] A. Egges, S. Kshirsagar, and N. Magnenat-Thalmann. A Model for Personality and Emotion Simulation. In *Proceedings of Knowledge-Based Intelligent Information & Engineering Systems (KES2003)*, Lecture Notes in AI. Springer-Verlag: Berlin, 2003.

[13] K. Perlin and A. Goldberg. Improv: A system for scripting interactive actors in virtual worlds. In *Proceedings of the* 23^{rd} *Annual Conference on Computer Graphics and Interactive Techniques*, pages 205–215, August 1996.

[14] A. Pina, E. Cerezo, and F. J. Serón. Computer Animation: From Avatars to Unrestricted Autonomous Actors (A Survey on Replication and Modelling Mechanisms). *Computer & Graphics*, 3(24):297–311, 2000.

[15] G. Anastassakis, T. Ritchings, and T. Panayiotopoulos. Virtual Agent Societies with the mVITAL Intelligent Agent System. In R. De Antonio, A. Aylett and D. Ballin, editors, *Proceedings of Intelligent Virtual Agents*, Lecture Notes in AI, pages 112–125. Springer-Verlag: Berlin, 2001.

[16] H. Noser and D. Thalmann. Towards Autonomous Synthetic Actors. In *Proceedings of French-Japan Workshop on Synthetic Worlds*, Aizu (Japan), 1993. http://ligwww.epfl.ch/~thalmann/papers.dir/autonomous.pdf (last accessed on May 2004).

[17] F. de Rosis, C. Pelachaud, I. Poggi, V. Carofiglio, and B. De Carolis. From Greta's Mind to her Face: Modelling the Dynamics of Affective States in a Conversational Embodied Agent. *International Journal of Human-Computer Studies*, 59:81–118, 2003.

[18] R.A. Brooks. Intelligence Without Representation. *Artificial Intelligence*, 47:139–159, 1991.