Scripting Affective Communication with Life-like Characters in Web-based Interaction Systems

Helmut Prendinger Sylvain Descamps Mitsuru Ishizuka
Dept. of Information and Communication Engineering
Graduate School of Information Science and Technology
University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
{helmut,descamps,ishizuka}@miv.t.u-tokyo.ac.jp

Abstract

In this paper, we discuss scripting tools that aim at facilitating the design of web-based interactions with animated characters capable of affective communication. Specifically, two systems are developed. The SCREAM system is a scripting tool that enables authors to create emotionally and socially appropriate responses of animated characters. Content authors may design the mental make-up of a character by declaring a variety of parameters and behaviors relevant to affective communication and obtain quantified emotional reactions that can be input to an Animation Engine. While the default operations of a character's 'mind' are based on psychological and sociological research, authors may easily modify and extend its rule set. In order to facilitate high-level scripting and connectivity with other web-based animated agent systems, the tool is written in a lightweight Java based Prolog system and Java. Connectivity is demonstrated by interfacing SCREAM with the our second system, MPML, an XML-style markup language that allows to control and coordinate the multi-modal appearance of synthetic characters. Affective communication with animated characters is illustrated by the implementation of a web-based casino scenario.

Submitted to: Applied Artificial Intelligence Journal

Statement: The authors certify that this work has not been published elsewhere and that it has not been submitted simultaneously for publication elsewhere.

Abbreviated title: "Scripting Affective Communication with Life-like Characters" or if too long, "Scripting Affective Communication"

1 Introduction

Recent years show a growing interest in animated characters as conversational partners for a variety of tasks that are typically performed by humans. Among others, animated characters are used as virtual *tutors* in interactive learning environments (de Rosis et al., 1999; Johnson et al., 2000; Hayes-Roth, 2001), as virtual *presenters* (André et al., 2000; Cassell et al., 2000; Ishizuka et al., 2000), and as virtual *actors* for entertainment (Rousseau and Hayes-Roth, 1998; Paiva et al., 2001) and language conversation training (Prendinger and Ishizuka, 2001c,b). Sometimes, the umbrella term 'Virtual Personal Service Assistants' is used to refer to the class of animated interface agents (Arafa and Mamdani, 1999).

Key problems in the design of animated characters are to make them believable in their respective roles and life-like in their overall behavior. If successful, those characters can be pedagogically effective as tutors, convincing as presenters, and dramatically interesting as actors. While there is general agreement that emotion and personality are mandatory ingredients of life-like characters, no consensus exists over which other mental concepts should be modeled to drive their behavior. Most researchers endow characters with some additional features such as the planning (Gratch, 2000) or awareness of the social context (Prendinger and Ishizuka, 2001c; de Carolis et al., 2001). Allen (1999), on the other hand, considers a broad range of 'higher-level' mental concepts—personality, attitudes, standards, moods, emotions, desires, intentions, and plans—which he calls (motivational) 'control states'. A mental concept in Allen's sense is considered as a control state if it might function as a predictor of behavior. In this paper, we will propose a general framework that covers the most important concepts for emotion-based characters and has the flexibility to extend or modify the class of concepts, rather than arguing for a definite set of control states.

Believability and life-likeness are also strongly associated with a character's visual appearance (McBreen et al., 2000) that is often based on sequences of 2D cartoon drawings or a 3D body motion or facial model. Since we are primarily interested in web-based distribution of interactive characters, we opted for easy-to-use 2D cartoon-style characters that can be controlled with the Microsoft Agent package (Microsoft, 1998). Those characters may utilize several communicative modalities, such as facial displays, gestures, and synthetic speech.

Currently, the success of most of the afore mentioned systems relies on the careful crafting of their designers, who are typically programmers. We believe that the growing popularity of animated agent systems will increase the demand for tools that allow content experts rather than programmers to script interactive affective behavior of animated characters in a simple and intuitive way. For this purpose, we have developed two scripting tools, one of which deals with the mental processes of a character, while the other is responsible for coordinating the embodied behavior of one or more characters.

Our SCREAM (**SCR**ipting **E**motion-based **A**gents **M**inds) system is a tool to script a character's mind. It is intended as a plug-in to content and task specific agent systems such as interactive tutoring or entertainment systems that provide possible verbal utterances for a character. Our system may then decide on the kind of emotion expression and its intensity, based on a multitude of parameters that are relevant to the current interaction situation. Parameters are derived from the character's mental state as well as the peculiarities of the social setting in which the interaction takes place, and features of the character's interlocutor(s), e.g., the user. While the nature of an (appropriate) affective reaction is obvious in many cases, more complex interaction scenarios will likely face the problem of 'conflicting emotions'. Consider a situation where an animated teammate that is very angry and extrovert interacts with a new team member in the course of an important mission. The 'conflict' here consists in the fact that the character's display of its anger might increase the insecurity of the new member and thereby endanger the success of the mission. Our system provides various controls to ensure situationally adequate behavior consistent with a character's mental makeup. Characters are adaptive in the sense that affective features of the interaction history result in updated values for certain mental states, such as attitudes and social relations.

As a scripting tool, the SCREAM system can easily be extended by adding or modifying rules that encode the character's cognitive processes such as appraisal, or intensity combination functions. An important feature of our system is the *granularity* of scripting character behavior. At the simple end of the spectrum, the author (content provider) controls each dialogue contribution of a character. While this may be feasible for single-character presentation tasks, interaction scenarios require that characters decide their behavior autonomously. A character's mental state can be designed at many levels of detail, from driven purely by (personality) traits to full awareness of the social interaction situation including character-specific beliefs and beliefs attributed to interacting characters. The flexibility of our approach to scripting agent minds allows to create agent personalities with varying degrees of adaptivity and sophistication.

Our MPML (**M**ultimodel **P**resentation **M**arkup **L**anguage) tool is used to script animated character scenarios. It is an XML-style markup language that allows content authors to synchronize animation and synthetic speech of multiple characters. Besides tagging schemes for the control of characters' embodied behavior, the markup language provides a tagging scheme to process speech input from a user. MPML is typically used to easily generate scripted behavior of animated characters, as specified by the content author. Furthermore, via an interface to the SCREAM system, MPML may execute a character's behavior as suggested by its mind.

The rest of the paper is organized as follows. The next section reviews some of the literature on scripting the bodies and minds of animated characters. Section 3 offers a step-by-step introduction to the core components of the SCREAM system architecture, emotion generation, emotion regulation, emotion expression, and affect processing. In Section 4, the MPML tool and its interface to the SCREAM system will be described. Section 5 demonstrates the SCREAM system by an extended example of affective interaction in a web-based casino. In Section 6, we discuss and conclude the paper.

2 Related Work

Scripting tools for animated characters can be broadly categorized into tools that deal with a character's 'body', the *Animation Engine*, and those that concern the character's 'mind', the *Behavior Engine* (Perlin and Goldberg, 1996). In the following, we will review some of those attempts, with a focus on tools for scripting a character's affect-related states and processes.

2.1 Scripting Characters' Bodies

Animating the visual appearance of characters is a difficult task that involves many levels, from changes to each individual degree of freedom in a character's motion model to high level concerns about how to express a character's personality by means of its movements.

The Improv system of Perlin and Goldberg (1996), e.g., allows scripting continuous agent motions and smooth transitions between them. Many systems synchronize lip shapes to synthetic speech by extracting 'visemes' from typed text. Recently, Cassell et al. (2001) introduced the **B**ehavior **E**xpression **A**nimation **T**oolkit (BEAT) that facilitates synchronization of non-verbal (conversational) behavior and synthetic speech. Badler et al. (2000) developed a series of powerful animation engines, also considering authoring issues (e.g., the Jack Presenter). One scripting level higher is our MPML tool that allows for easy control of pre-defined motion sequences of a character, as well as the synchronization of the behavior of multiple characters (Ishizuka et al., 2000; Descamps et al., 2001).

2.2 Scripting Characters' Minds

A task complementary to scripting a character's (visual) animation is to script a character's 'mind', e.g., the way a character perceives its surrounding (and internal) world and how the agent reacts emotionally. A direct way of scripting a character's responses is realized by the 'Verbots' of Virtual Personalities (Verbot, 1998). An author may specify rules (including wild cards) that fire upon certain patterns of typed texts, thereby allowing Eliza-style conversations. For a similar purpose, the **A**rtificial **I**ntelligence **M**arkup **L**anguage (AIML) has been developed (Ringate, 2001). However, since neither the Verbot nor the AIML approach support the definition of an emotion or world model, they are of limited use for scripting emotionally adequate and consistent agent behavior.

Elliott (1992) describes a full-fledged architecture for reasoning about emotion and personality, the Affective Reasoner. Despite its importance as a simulation platform for emotion-based agent interactions, the Affective Reasoner does not (directly) address the scripting issue. The Em architecture of Reilly (1996) consists of a set of scripting tools for the creation of emotion-based agents. Emotion generators (a kind of appraisal rules) take sensory inputs and produce so-called emotion structures, which are emotion types together with information about intensity, cause, and direction (e.g., "happy for Lisa with intensity 4 because she won in the lottery"). The resulting emotion structures are processed by combination and decay functions and are then mapped into behavioral features, i.e., instructions about the agent's behavior. Authors can build agents with Reilly's Em by means of so-called Hap rules that determine input and output of the respective processes. Our system is in many ways similar to the EM architecture, but more flexible in the sense that it allows scripting at various abstraction levels. Furthermore, our system exploits web technologies so that emotion-based synthetic characters can be run in a web browser. Blumberg (1996) proposes an ethology-inspired approach to interactive agents, and focuses on the problem of action selection in artificial animals ('animats'). Hamsterdam, an object-oriented toolkit, allows to build autonomous animats by defining their internal needs, activities they can perform, a sensory and motor system, and by using a behavior system that implements Blumberg's model of action selection. However, action selection is based on animal-specific drives and motivations rather than an explicit emotion model.

The work of Arafa and Mamdani (1999) on Virtual Personal Service Assistants describes requirements of agents in real-time multi-agent systems. Specifically, they propose an agent communication language called *Asset Description Language* (ADL) that may contain affective markups of the conveyed information. Although their and our work focus on different aspects of affective communication, the overall architecture shares many similarities. The probably most powerful authoring system for web-based user-agent interactions is the commercial toolkit developed by Extempo Systems Inc. (Ext, 2001), based on

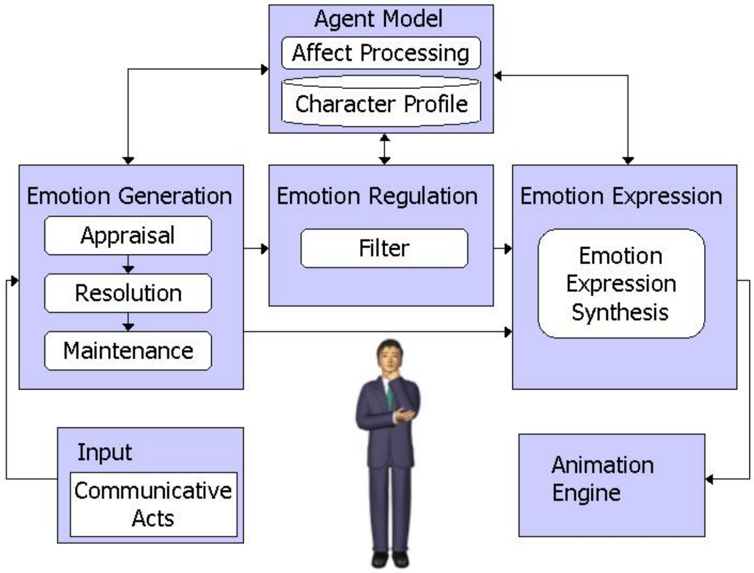

Figure 1: SCREAM System Architecture.

Hayes-Roth's experience with interactive animated characters (Rousseau and Hayes-Roth, 1998; Hayes-Roth, 2001).

On a yet higher scripting level, André et al. (2000) describe a mechanism that allows to automatically design presentation dialogues between multiple animated characters. The approach is plan-based and conditions characters' responses on their role in the scenario and models of emotion and personality. The JAM agent architecture (Huber, 1999) used as a tool to encode the planning component constitutes a rich agent programming language. Although our system provides support for authoring character ensembles, it does not do so automatically. The main reasons are that we wanted to give the author control over each individual dialogue move and that we cannot rely on a database with product entries and associated attributes which can be evaluated against the agents' goals and interests.

3 The SCREAM System

The SCREAM system for scripting emotion-based agent minds is written in Java for portability. In order to support high-level scripting of a character's mind, we use Jinni 2000, a Java based Prolog system (BinNet, 2000). Basically, a character's mind contains a user-extensible set of rules and facts. Scripting characters in a declarative language like Prolog is close to the "English-style" scripting language proposed by Perlin and Goldberg (1996). Fig. 1 gives an overview of the system.

The following sections offer a walk through the main components of the SCREAM system: Emotion Generation, Emotion Regulation, Emotion Expression, and the Agent Model that is also responsible for updating a character's mental state. We will start by specifying the input to the agent system.

3.1 Input to a Character's Mind

Emotion-based agents receive input in the form of communicative acts of the form com_act(S,H,Concept,Modalities,Sit)

where S is the speaker (locutor-agent), H the addressee (interlocutor), *Concept* the information conveyed by S to H in situation *Sit*, and *Modalities* is the set of communicative channels used by S. Besides specific facial displays, gestures, and posture, communicative modalities may also include information about acoustical correlates of (expressed) emotions and linguistic style. Of course, the employed modalities will crucially depend on the characteristics of the Animation Engine used—what the animated character is able to express—as well as the interlocutor. For the special case where a character interacts with a human, communicative modalities are typically difficult to recognize (Picard, 1997) and hence a user's affective state is hard to infer (Ball and Breese, 2000).

Most importantly, communicative acts have preconditions that must be declared. The question "You don't wanna trick me, do you?", uttered by a dealer in a casino might have the following preconditions:

wants(dealer,player,fair_play,3,s15)
blameworthy(dealer,player,not_fair_play,5,s15)

That is, the dealer has the goal of a fair play, and considers it as blameworthy if the player does not respect the rules of the game. The numbers "3" and "5" refer to intensities and will be explained in the next section. Since our system lacks a language understanding module (let alone an affective language understanding module), a character has to receive those propositions in order to model its interlocutor's affective state. A special case of a communicative act is a simple act of the form act(O,Event,Sit) with O the observer that experiences *Event* in situation *Sit*, e.g., when a new actor appears on stage.

3.2 Emotion Generation

A core activity of an emotion-based agent mind is the generation and management of emotions, which is dealt with by three modules, the *appraisal* module, the *emotion resolution* module, and the *emotion maintenance* module. Each of them will be described in detail in the following sections.

3.2.1 Appraisal

According to a widely used definition, an agent's appraisal refers to the process that qualitatively evaluates (external and internal) events according to their emotional significance for the agent (Ortony et al., 1988; Elliott, 1992; Reilly, 1996; Gratch, 2000; Ortony, 2001). Emotions are seen as valenced (i.e., positive or negative) reactions to events including other agents' actions (or the agent's own actions) and the perception of objects, qualified by the agent's goals, standards, and preferences. In Ortony's words "People only get into emotional states when they *care* about something [...]—when they view something as somehow good or bad." (Ortony, 2001).

In computational models of emotion the significance of an event is determined by so-called 'emotion-eliciting conditions' (EECs), which comprise an agent's relation to four types of abstract mental concepts.

- *Beliefs.* States of affairs that the agent has evidence to hold in the (virtual) world.
- *Goals.* States of affairs that are (un)desirable for the agent, i.e., what the agent wants (does not want) to obtain.

- *Standards.* The agent's beliefs about what ought (not) to be the case, i.e., events that the agent considers as praiseworthy (blameworthy).

- *Attitudes.* The agent's dispositions to like or dislike other agents or objects, i.e., what the agent considers (not) appealing.

In this paper, an agent's associated mental states are uniformly treated as 'propositional attitudes' that are conceived as relations between the agent and some abstract concept, the content of the attitude (Barwise and Perry, 1983).

According to the emotion model of Ortony et al. (1988), which is commonly known as the OCC model, emotion types are just classes of eliciting conditions, whereby each is labelled with an emotion word or phrase. In total, twenty-two classes of eliciting conditions are identified, including *joy*, *distress*, *happy for*, *gloat*, *resent*, *sorry for*, *reproach*, and *angry at*. Among the emotions with the simplest specification is the well-being emotion *distress* (written in Prolog-style form close to actual code, see also O'Rorke and Ortony (1994)).

$$\text{distress(L1,L2,F,}\delta\text{,Sit)} \textbf{ if } \text{wants(L1,F,}\delta_{Des(F)}\text{,Sit)} \textbf{ and } \text{holds(non-F,Sit)}$$

An agent $L1$ may experience distress over some state of affairs (or 'fluent') F in a situation Sit if $L1$ wants a fluent that does not hold in Sit. The second argument in the proposition $distress(L1, L2, F, \delta_{Des(F)}, Sit)$ denotes the (possibly not specified) agent 'toward' which the agent is distressed. Although the standard specification of *distress* does not involve an addressee, we consider all emotions as genuinely social, and at least directed to the user if no other agent is present. Later on, this feature will be motivated in more detail in the context of the expression of an emotion. The fourth argument in the proposition refers to the intensity δ of the emotion. Emotions are not simply present or absent, they have varying intensities depending on the degree to which, e.g., a state of affairs is desirable to the agent. For all mental states, we uniformly assume intensity degrees $\delta_i \in \{0, \ldots, 5\}$ such that zero is the lower threshold (the state is not active), and five is the maximum intensity (values greater than five are mapped to five). For simplicity, we ignore the degree to which an agent believes that a proposition is true, e.g., that the opposite of F holds. In the case of *distress*, δ is set to $\delta_{Des(F)}$.

For many emotion types, intensities have to be combined, as their specification involves multiple mental concepts. For instance, the emotion type *angry at* depends on the agent's standards in addition to its goals.

$$\begin{aligned}\text{angry-at(L1,L2,F,}\delta\text{,Sit)} \textbf{ if } & \text{holds(did(L2,A),Sit)} \textbf{ and } \text{causes(A,F,Sit}_0\text{)} \textbf{ and} \\ & \text{wants(L1,non-F,}\delta_{Des(non\text{-}F)}\text{,Sit)} \textbf{ and} \\ & \text{blameworthy(L1,A,}\delta_{NAcc(A)}\text{)} \textbf{ and } \text{Sit}_0 < \text{Sit}\end{aligned}$$

By default, intensities δ_i are combined to an overall intensity degree δ by logarithmic combination, such that $\delta = \log_2\left(\sum_i 2^{\delta_i}\right)$. If an agent desires the opposite of F with intensity 3 and considers the other agent's action blameworthy with 'non-acceptability' degree 2, the agent is angry at the other agent with intensity 4 (rounded value).[1]

[1]Our method to combine the intensities associated with mental states is certainly simplistic, and has been criticized by Fiorella de Rosis (personal communication) on grounds that intensities applying to such different entities as, e.g., the desirability of goals and the appealingness of an object should not be lumped together. In order to solve the problem *formally*, we adopted a situation theoretical viewpoint (Barwise and Perry, 1983) that uniformly treats all relevant mental states as propositional attitudes with associated intensities. We have currently no *empirical* evidence (beyond our intuition) that people combine intensities the way we suggest. In any case, content experts using our system are free to use their own combination rules.

Modeling the Appraisal of Others

An interesting subset of the OCC emotion types—the so-called fortunes-of-others emotions—assumes that an agent is able to attribute emotions to other agents (Elliott and Ortony, 1992). Consider the emotion specification of the *happy for* emotion type.

$$\mathsf{happy\text{-}for(L1,L2,F,}\delta\mathsf{,Sit)}\ \textbf{if}\ \mathsf{likes(L1,L2,}\delta_{App(L1,L2)}\mathsf{,Sit)}\ \textbf{and}\ \mathsf{joy(L2,F,}\delta_{Des(F)}\mathsf{,Sit)}$$

Here, the proposition $joy(L2, F, \delta_{Des(F)}, Sit)$ denotes the agent's *belief* (or hypothesis) that agent $L2$ experiences *joy* over fluent F with intensity degree $\delta_{Des(F)}$. The intensity of the *happy for* emotion is then obtained by logarithmic combination of the degree with which the agent likes the other agent and the hypothesized *joy* of the other agent. The fortunes-of-others emotion types (*happy for*, *gloating*, *resent*, and *sorry for*) pose the difficult problem of *assessing* the emotional states of other agents (interlocutors). There are at least three ways to approach the problem.

- *Designer's definition.* Emotional states of other agents can be pre-defined by the designer of the interaction scenario, as part of the definition of communicative acts (see above).
- *Stereotypes.* Assuming that other agents are correctly classified, stereotypes (Rich, 1979) can be employed to derive their features, e.g., a typical visitor of a casino may be assumed to have the goal of winning money, and thus be joyful if she or he wins.
- *Emotion recognition.* Given other agents' expression of emotion, their emotional state can be inferred from communicative modalities, such as facial displays, prosody, linguistic style, posture, and so on (Picard, 1997; Ball and Breese, 2000).

The problem is considerably simplified by the fact that in the case of the fortunes-of-others emotions, the only emotions to be inferred or recognized are *joy* and *distress*. In our current implementation, a hybrid approach is employed to assess the user's emotional state where her or his social role (determined by the interaction scenario) is associated with certain goals and default intensity values—which allows to infer the two mentioned well-being emotions—and a surface analysis of the user's linguistic style categorizes her or his utterance as friendly or unfriendly. In the case of animated interlocutors, mental states are defined by the designer of the scenario. Observe that in both cases, the very same emotion type specifications can be used to infer emotional states.

Mental states are not only determined by an agent's (social) role but also by its personality, e.g., friendly agents tend to react positively and have positive attitudes toward other agents, and extrovert agents are more likely to express their emotions. In the following section, we will discuss our concept of personality.

Personality

We will adopt the characterization of personality suggested by Moffat (1997, p.133):

> "**Personality** is the name we give to those reaction tendencies that are consistent over situations and time."

Although the literature on the definition of personality is highly controversial, it is generally seen as an affective state that is long in duration and not focussed on a particular event, which makes personality clearly distinguishable from emotions that are short-lived and focussed. Thus, personality will be conceived as a biasing mechanism for emotion generation and later on, for emotion expression. As convincingly argued by Rousseau and Hayes-Roth (1998), consistency of an agent's behavior is of paramount importance for its believability, and well-defined personality traits may guarantee consistency.

In the context of this paper, we will focus on two dimensions of personality, that are considered as crucial for social interactions between agents (André et al., 1999; Arafa and Mamdani, 1999).

- *Agreeableness* refers to an agent's disposition to be sympathetic (express 'positive' emotions, suppress 'negative' ones): friendly, good-natured, forgiving.
- *Extraversion* refers to an agent's tendency to take action and express emotions: sociable, active, talkative, optimistic.

In our model, we assume numerical quantification of personality dimensions, with a value $\gamma \in \{-5, \ldots, 5\}$. By way of example, a value of 3 in the agreeableness dimension means that the agent is rather friendly, whereas the value 3 in the extraversion dimension describes a rather introvert agent. Although a zero value is provided formally, we explicitly discourage character creators to design agents with 'neutral' personality (for any dimension). First, characters that are neutral in some respect tend to be less interesting and lack (dramatic) impact and secondly, many of the agent's mental states and behaviors can be explicitly conditioned on its personality, such as goals, attitudes, the decay rate of emotions, and tendency to express an emotion in a conversation. An agent's personality is stated as facts of the form

personality-type(L,agreeableness,γ_A)
personality-type(L,extraversion,γ_E)

Other personality dimensions (e.g., *neuroticism*) can be easily added in a similar fashion.

3.2.2 Emotion Resolution

A reasonably interesting agent will have a multitude of (possibly 'conflicting') emotional states at the same time, given the diversity of its goals, attitudes, and standards. The emotions generated by the appraisal process in a given situation *Sit* will be called *active* emotions (in *Sit*) and are collected together with their in intensities in a set $\{\langle E_1, \delta_1, Sit\rangle, \ldots, \langle E_n, \delta_n, Sit\rangle\}$. The key question here is which emotion the agent will most likely express (or 'have'), if e.g., it is joyful with intensity 4, happy for another agent with intensity 3, and also distressed with intensity 3?

Our approach to emotion resolution assumes that emotions are partitioned into *positive* emotional reactions and *negative* emotional reactions (Reilly, 1996; Ortony, 2001). More precisely, we might distinguish between 'benevolent' and 'malevolent' emotions in order to emphasize the communicative implications of (the expression of) emotions, i.e., as positive or negative reactions towards other agents. However, certain emotions are not necessarily directed to other agents (e.g., agents can be *joyful* by themselves). Examples of positive emotions are *joy*, *happy for*, and *sorry for*, whereas *resent* and *angry at* are negative emotions.

In our approach, the presence of multiple emotions is resolved by computing and comparing two affective states.

- The *dominant emotion type* is simply the emotion with the highest intensity value. If there is no unique dominant emotion, it is decided by the agent's personality (see below).

- The *dominant mood type* is calculated by considering all active emotions. Let δ_i^+ denote the intensity value of an emotion in the set of (active) positive emotions, and δ_i^- the intensity of an emotion of the (active) negative emotions. We compare the values of

$$\Delta^+ = \log_2 \left(\sum_i 2^{\delta_i^+} \right) \quad \text{and} \quad \Delta^- = \log_2 \left(\sum_i 2^{\delta_i^-} \right)$$

  and determine the dominant mood by the higher value. The agent's personality will be consulted if no unique highest value exists.

The *winning emotional state* is then obtained by comparing the intensities for dominant emotion and dominant mood. In effect, we can account for situations where an agent has a high-intensity positive emotion but is still more influenced by its overall negative mood. In situations where equal intensities (of active emotions or mood) result, the agreeableness dimension of an agent's personality is employed. In line with our concept of personality as a biasing mechanism, an agent with disagreeable personality will favor, e.g., a winning negative emotional state to a positive mood state if both have the same intensity level.

3.2.3 Emotion Maintenance

Once generated, emotions typically have a rather short lifespan in the agent's mind. Emotion maintenance handles the decay process of emotions. A decay function determines how fast the intensity level of the active emotions decreases each 'beat'. Following Mateas and Stern (2000), a *beat* is defined as a single action-reaction pair between two agents or in the case of more than two agents, a cycle until the agent takes initiative again. When the intensity level is less or equal to zero, the corresponding emotion is removed from the set of active emotions. The decay rate is determined by the emotion type and the agent's personality.

decay-rate(L,E,2) **if** member(E,Negative-Emotions)) **and**
personality-type(L,agreeableness,γ_A) **and** $\gamma_A > 0$

This rule says that if the agent is agreeable (i.e., $\gamma_A > 0$) the intensity level of negative emotions is decreased with rate 2. Similarly, the decay rate of positive emotions of an agreeable agent can be set to 1. Two notable exceptions in the decay process are the emotions *hope* and *fear* that are removed from the set of active emotions when the relevant state of affairs responsible for the agents' hope or fear is added or removed, respectively (Reilly, 1996).

3.3 Emotion Regulation

In social contexts, an agent's (winning) emotional state is not always expressed with the intensity derived from emotion generation. There is ample evidence in the literature that conversing agents *regulate* (e.g., suppress) their emotions to adapt to the conditions of a particular social situation. Recently, the mechanisms underlying emotion regulation received considerable interest in psychology. Gross (1998, p.275) gives the following characterization:

> "Emotion regulation refers to processes by which individuals influence which emotions they have, when they have them, and how they experience and express these emotions."

Specifically, Gross distinguishes five kinds of emotion regulatory processes: (i) *situation selection*, e.g., approaching or avoiding people, (ii) *situation modification*, i.e., tailor a situation in order to modify its emotional impact, (iii) *attention deployment*, i.e., select an aspect of a situation, (iv) *cognitive change*, i.e., selecting one of the possible interpretations of a situation, and (v) *response modulation*, i.e., influencing the response tendencies once an emotion has been elicited. While most of the emotion regulatory processes can be integrated to a more elaborate theory of appraisal, we will concentrate on the last kind of regulatory process that refers to the modulation of emotion-expressive behavior.

It is important to mention that emotion regulation only applies to emotional reactions that can be assumed to be controlled by the agent. Some reactions are involuntary and spontaneous (Ortony, 2001), and hence not subject to emotion regulation. For instance, while an agent may control (suppress) its verbal expression of anger, it might still show somatic reactions such as shaking or flushing.

3.3.1 Parameters for Emotion Regulation

The phenomenon of emotion regulation as response modulation is discussed in different fields that suggest a variety of regulation parameters.

- *Semiotics.* Ekman and Friesen (1969) argue that the expression of emotional states (e.g., as facial expression) is governed by social and cultural norms—so-called *display rules*—that have a significant impact on the intensity of emotion expression.

- *Speech act theory.* Recognizing that a significant part of human conversation takes place in social (or socio-organizational) settings, Moulin (1998) describes the impact of participating agents' social roles on their communicative behavior. Besides behavioral constraints associated with a role (responsibilities, duties, rights, prohibitions, and possibilities), Moulin observes that agents obey communicative conventions specific to their role (see also Lewis (1969)). These conventions serve as a regulatory for the agent's choice of verbal expressions in a given context.

  In their work on politeness and social interaction, Brown and Levinson (1987) discuss social parameters that determine the linguistic style of an agent's response. A computational account of their theory is given in Walker et al. (1997).

- *Affective Communication.* In the area of affective communication with animated characters, some authors integrate models of emotion regulation to emotion-based agent architectures. While Prendinger and Ishizuka (2001c) focus on the agent's personality and its social role as regulatory parameters, de Carolis et al. (2001) discuss a wide variety of emotion modulating factors, including the nature of the emotion in question (e.g., emotion valence, social evaluation of an emotion), and scenario factors such as the agent's display motive, its personality, features of the interlocutor (e.g., personality), the role relationship to the other agent, and the type of social interaction (private or public).

A computational account of emotion regulation has to combine parameters that influence the emotion regulation process. However, it is not obvious how this can be done in an psychologically adequate way. In the following, we will propose a set of (weighted) parameters together with default combination functions. We opted for a categorization into parameters that can change over time, the social variables, and parameters that pertain to the agent's (ability of) self-control and the current interaction context.

Social Variables

When agents interact, they do not only exchange information but also establish and maintain social relationships. Hence it is important that agents avoid introducing disharmony into a conversation (Moulin, 1998) or threaten other agents' public face (Walker et al., 1997). We assume that emotion expression (e.g., facial display or linguistic style) is determined by personal experience, background knowledge, and cultural norms (Walker et al., 1997), as well as the 'organizational culture' (Moulin, 1998). Following Brown and Levinson (1987), we take *social power* and *social distance* as two important parameters determining the agent's regulation of emotion. They also consider the ranking of imposition of different speech acts as a social variable, which is currently ignored in our approach since utterances are not explicitly framed as speech acts. Two social relations are defined for agents $L1$ and $L2$.

social-power(L2,L1,θ_P,Sit)
social-distance(L1,L2,θ_D,Sit)

The first (asymmetric) relation says that agent $L2$ is θ_P ranks higher than $L1$ in Sit, as perceived by $L1$. The second (asymmetric) relation refers to the closeness between two agents, whereby lower values indicate more closeness, as perceived by $L1$ ($\theta_P, \theta_D \in \{0, \ldots, 5\}$). Observe that Cassell and Bickmore (2001) use the term "familiarity" for what we call "distance", and interpret social distance as solidarity or 'like-mindedness'. In our case, social distance is simply taken as a indicator how 'freely' an agent may express its emotions without social sanction. Based on θ_P and θ_D, agent $L1$ computes the *social threat* from $L2$ as $\theta = \log_2 \left(2^{\theta_P} + 2^{\theta_D}\right)$, whereby θ is set to zero if θ_P and θ_D are both zero. Besides logarithmic combination, other combination methods are used in the literature, e.g., Walker et al. (1997) simply add the values of the social variables. Note that in our view social variables are not meant to reflect 'objective' ratings of power and distance, but the modelled agent's assessment of the ratings.

Control Variables

The following parameters describe an agent's inclination to (self-)control its negative emotions (e.g., given its personality traits) as well the agent's inclination to suppress negative emotions as a result of external factors of the interaction scenario (e.g., the other agent's personality). The following list of control variables is meant as an important subset of the more complete set of conceivable variables discussed by de de Carolis et al. (2001).

- *Personality.* An agent is more likely to express negative emotions if it is either disagreeable or extrovert.
- *Other agent's personality.* If the agent assumes that the interlocutor's personality is disagreeable, it will rather not express a negative emotion. For instance,

personality-type-other(L,agreeableness,4) states that L is considered very disagreeable by the modelled agent (note that signs are reversed).

- *Reciprocal feedback.* An interesting phenomenon in human-human interaction are reciprocal feedback loops where one agent's linguistic friendliness results in the interlocutor agent's adaption to its otherwise unfriendly behavior. Similarly, there are 'negative' feedback loops where agents adjust to linguistic unfriendly behavior. The proposition

  linguistic_style_other(L,friendly,5)

  expresses that the modelled agent is motivated to rather suppress negative emotions as the interlocutor L chooses very friendly linguistic style.

The overall control value $\gamma \in \{-5, \ldots, 5\}$ is computed as

$$\gamma = \frac{\sum_i \frac{\gamma_i}{IMP_i}}{N}$$

where the denominator N scales the result according to the number of considered control parameters. Since some system users might also want to weight the impact IMP of each parameter, we support the use of $\frac{\gamma_i}{IMP_i}$ and default IMP_i to the value one. In essence, the formula captures the intuition that different control parameters may defeat each other. Thus, the control of an agent that is very extrovert but deals with a very unfriendly other agent will be neutralized. Further control parameters can be easily added to (or deleted from) our model, such as other interlocutor features like display motive or the interlocutor's cognitive capacity (de Carolis et al., 2001).

3.3.2 Combining Regulation Parameters

At this point we are given the winning emotional state as well as the regulation parameters social threat and control. In order to determine the intensity of the *external emotion*, i.e., the type of emotion to be displayed by the character, the intensities of emotional state and regulation parameters have to be combined. In the following, we will propose two combination methods. The first method consists of a straightforward linear combination of intensities, whereas the second one casts emotion regulation as a decision problem.

Linear Combination

A so-called *filter program* (Prendinger and Ishizuka, 2001c) consists of only one rule with two different combination functions, one for positive and one for negative emotions.

external-emotion(L1,L2,E,ϵ,Sit) **if** social-threat(L1,L2,θ,Sit) **and**
control(L1,L2,γ,Sit) **and**
winning-emotional-state(L1,L2,E,δ,Sit)

The combination functions are as follows:

- *Negative emotions.* The intensity of the external emotion ϵ is obtained by $\epsilon = \delta - (\theta + \gamma)$. Hence, the function balances social threat against the character's control, whereby high values for threat may neutralize the lacking (self-)control of the character to a certain extent.

- *Positive emotions.* Here, the intensity level results from calculating $\epsilon = \delta - (\theta - \gamma)$ and therefore, a high level for threat or a negative value for control will decrease the intensity of a positive external emotion. A positive value for control may defeat the impact of social threat.

A filter program constitutes an easy and practical method to determine whether and to what extent a character should suppress an emotion (Prendinger and Ishizuka, 2001c). However, there are situations where the character's expression (or suppression) of an emotion has a more specific meaning, such as (not) to induce an emotion in its interlocutor. For instance, a character may want to make its interlocutor feel the same emotion. Here, the *display motive*, the reason to express an emotion would be empathy which calls for a more powerful combination method de Carolis et al. (2001) for a more complete list of display motives).

The next section briefly introduces decision theory as a framework to determine the intensity of emotion expression. Despite its promise as a technique for inferring behavior tendencies in emotion-based agents (Gmytrasiewicz and Lisetti, 2000), we only use it as an add-on to our system considering the complexities of encoding and tuning probabilistic knowledge for non-experts.

Combination Using Decision Theory

A *decision network* (sometimes called *influence diagram*) graphically represents a decision problem. More specifically, it is a representation of an agent's belief state, actions it may carry out and the states resulting from those actions, as well as the utility of the resulting states (Russell and Norvig, 1995; Poole et al., 1998). A decision network contains three types of nodes that will be explained by the example network shown in Fig. 2.

- *Chance nodes* (drawn as ovals) represent random variables just as they do in belief networks. All chance nodes have probability tables associated with them. Their parent nodes represent the variables that directly influence the variable corresponding to the chance node. In Fig. 2, *control* (the agent's self-control tendency), *social threat* and *negative emotion* (the type of the elicited emotion) influence *display tendency* (the agent's probability to express a negative emotion).

- *Decision nodes* (drawn as rectangles) represent decision variables and refer to action choices of the agent, such as *check negative consequences* (of displaying a negative emotion).

- A *utility node* (drawn as a diamond) represents the agent's utility function.[2] There is only one utility node, and its parents represent states that affect utility. In our example, the variables *negative emotion*, *check for negative consequences* and *suppress negative emotion* determine the utility for the agent.

The Negative Emotion Expression Problem of Fig. 2 is a sequential decision problem consisting of a sequence of two decisions. First, the agent decides whether to check for negative consequences of displaying a negative emotion, and then it decides whether to suppress the negative emotion. In a decision theoretic framework, for an agent to act rational means that it tries to maximize the expected utility of an action (in the case a

[2]Utility nodes are also called *value nodes* (Poole et al., 1998).

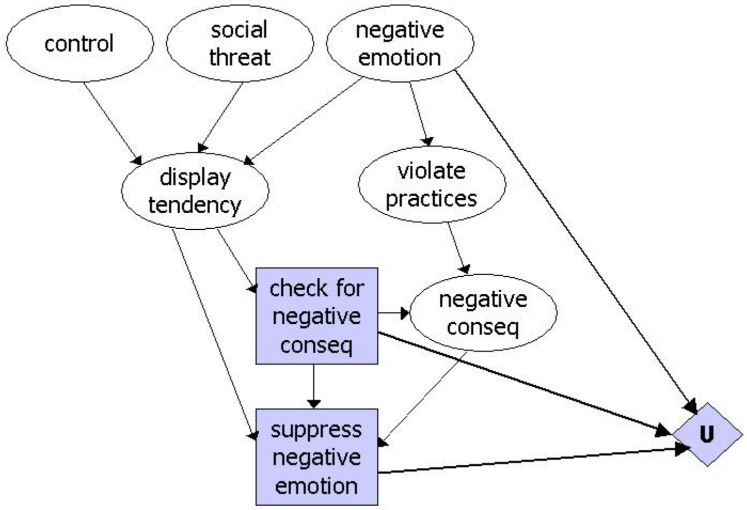

Figure 2: Decision network for the Negative Emotion Expression Problem.

simple decision problem) or a policy (in case of a sequential decision problem). Without going into any further detail about calculating an optimal policy (Russell and Norvig, 1995; Poole et al., 1998), we will show how to obtain the *expected utility* of the optimal action α for a simple decision problem which is defined as

$$EU(\alpha) = \max_A \sum_{Sit'} U(Sit') \times P(Sit'|E, A)$$

such that for each available action A and possible outcome Sit' of A, the agent's belief that A will result in Sit' given E is represented by $P(Sit'|E, A)$ where E denotes the agent's current knowledge, and the utility (or desirability) of a situation is denoted by $U(Sit)$.

By way of illustration, let us look at the expected utility (value) of the decision to suppress a negative emotion, obtained from our actual implementation of the network shown in Fig. 2. In the case where the agent has a display tendency of a negative emotion, checks for negative consequences, identifies such consequences, and then suppresses its emotion, the utility is -15 (see Gross (1998) for the 'cost' of suppressing emotions). On the other hand, when the agent decides not to suppress its (negative) emotion under the same circumstances, the utility is -99.78, therefore the agent looses less when suppressing the negative emotion. In order to obtain an optimal policy for the Negative Emotion Expression Problem, the agent has to exhaustively consider all possible cases.

We conclude this section by referring to the work of Conati (2001) who uses decision theory for the design of intelligent educational agents. In her work, a tutor agent has to decide whether to give or refrain from giving a student advice, depending on its model of the student's affective state (see also Gmytrasiewicz and Lisetti (2000)).

3.4 Emotion Expression

External emotions (with associated intensities) are eventually instantiated to actual verbal and non-verbal behaviors. In this paper, we use a simplified version of Ortony's categorization of emotion response tendencies, and distinguish between expressive and information-processing responses (Elliott, 1992; Ortony, 2001).

- *Expressive responses* include somatic responses (flushing), behavioral responses (fist-clenching, throwing objects), and two types of communicative responses, verbal and non-verbal (e.g., frowning).
- *Information-processing responses* concern the agent's diversion of attention and evaluations (which will be discussed in the next section).

Ortony (2001) also discusses *coping strategies* such as trying to calm oneself down or developing a plan to prevent the re-occurrence of a certain situation. However, since no planner is integrated to our system, we cannot deal with emotional responses that extend over multiple beats, i.e., action-reaction pairs (but see the Émile system of Gratch (2000)).

Expressive responses in our interaction scenarios are limited by the 'animations' pre-defined for each of the cartoon-style characters. Since not all characters come with adequate animations for somatic, behavioral, or even non-verbal responses, we often rely on verbal responses to convey emotions. Besides linguistic style (Walker et al., 1997), we use acoustic correlates of the so-called 'basic emotions' (Ekman, 1992): *fear*, *anger*, *sadness*, *happiness*, *disgust*, and *surprise*.[3] While Ekman concentrates on emotions that have distinctive facial expressions, Murray and Arnott (1995) describe the vocal effects on the basic emotions. For instance, if a speaker expresses *happiness*, her or his speech is typically faster, higher-pitched, and slightly louder, whereas the speech of a speaker expressing *sadness* is slightly slower and low-pitched.

The synthesis of emotion expression behaviors is strongly dependent on the expressiveness of the Animation Engine used for character animation. In our system, emotion expression synthesis is done in the simplest possible way, by defining verbal and non-verbal behaviors for each triple consisting of agent, situation, and emotion-intensity pair. The Animation Engine currently used (Microsoft, 1998) only allows for rather crude forms of combining verbal and non-verbal behavior. Body movements (including gestures) may precede, overlap, or occur subsequently to verbal utterances. More sophisticated tools such as the BEAT system (Cassell et al., 2001) may extract linguistic and and contextual information from (author pre-defined) spoken text and suggest intonation in speech as well as appropriate eye gaze and gestures. Alternatively, the Goal-Media prioritizer described as part of a Behavior Planner (de Carolis et al., 2001) can be used to synthesize verbal and non-verbal signals.

3.5 Affect Processing

In this section, we will describe the agent model (or *character profile*) which comprises the determinants of processes related to the generation, regulation, and expression of emotion. In order to simplify the agent model, we assume that some of its features can be treated as *static* whereas the majority are *dynamic* (i.e., they may change in the course of interaction). Among the static features are the agent's personality and standards. While it is reasonable to treat an agent's personality as permanent (or at least changing very slowly), we also assume that the agent's assessment of its interlocutor's personality does not change during interaction. Instead, the interlocutor's personality profile has to be explicitly given to the agent.

[3]In order to avoid confusion and since there is only a limited number of comprehensive 'emotion words', we use *slanted* when referring to Ekman's basic emotions instead of *italics* for Ortony's emotion types.

The dynamic features of an agent include its goals, beliefs, attitudes, as well as social power and distance relationships. In this paper, we will focus on change of attitudes and social variables, whereas goals and beliefs are updated in a straightforward way. A surface consistency check deletes a goal or belief p if the negation of p is added in a later situation. Currently, neither a planner nor a belief revision theory (Alchourrón et al., 1985) is incorporated to our system.

An important aspect of a character's believability and life-likeness is the ability to change its emotional reactions depending on the 'affective interaction history' with another agent. Simply put, if some interlocutor triggers mostly positive (negative) emotions in the character, it might change its attitude toward the interlocutor and be biased to appraise the interlocutor's future communicative acts in a more positive (negative) way (Ortony, 1991). Furthermore, a character will change its social distance (or familiarity) to another agent as a consequence of emotions elicited with that agent (Pautler and Quilici, 1998). In the following, we will discuss a model of attitude change, as well as update rules that describe the change of the social variables "distance" and "power".

3.5.1 Attitude Change

> "[...] I shall try to make a case for the claim that in addition to values being an important source of emotions, emotions are an important source of values and, more specifically, they can be the source of value in schemas." (Ortony, 1991, p. 341)

We will take Ortony's statement as a starting point and focus on one interpretation of values—as positive or negative attitudes toward an agent or object (*liking* and *disliking*). In particular, Ortony suggests the notion of *(signed) summary record* to capture our attitude toward or dispositional (dis)liking of another person. This record stores the sign of emotions (i.e., positive or negative) that were induced in the agent by an interlocutor together with emotions' associated intensities. For instance, if the interlocutor elicits *distress* with intensity 2, *angry at* with intensity 1, and *joy* with intensity 5, the summary record of the agent will contain two values, a negatively signed value of 1 (3 divided by the number of situations), and a positively signed one of 1.67. In order to compute the current intensity of an agent's (dis)liking, we simply compare the (scaled) sum of intensities of elicited positive and negative emotions (δ^σ, $\sigma \in \{+,-\}$), starting in situation $Sit_0^{L,I}$, the situation when the interaction starts.[4] We will only consider the winning emotional state δ_w, i.e., the most dominant emotion, although in general, multiple emotions may be elicited in each situation. If no emotion of one sign is elicited in a situation, it is set to zero.

$$\delta^\sigma(Sit_n^{L,I}) = \frac{\sum_{i=0}^{n} \delta_w^\sigma(Sit_i^{L,I})}{n+1}$$

Positive values for the difference $\delta^+ - \delta^-$ indicate an agent's liking of an interlocutor and negative ones indicate disliking (in a certain situation). For simplicity, we assume perfect memory of elicited emotions, such that the intensity of past (winning) emotions does not decay. If the interlocutor's recent behavior is mostly 'consistent' with the agent's past experience (i.e., both have same sign), it is reasonable to update the overall intensity of the agent's attitude according to the equation above.

[4] Situations also have to be parameterized by the agent and interlocutor here, referring to time points when emotions are elicited in an agent L due to communication with interlocutor I.

Ortony (1991) also considers the more interesting case where an interlocutor the agent likes as a consequence of consistent reinforcement (suddenly) induces a high-intensity emotion of the opposite sign, e.g., by making the agent very angry. He suggests three types of reactions for an agent: (i) agent L is uncertain how to construct the summary record value, (ii) L updates the summary value by giving a greater weight to the inconsistent information, or (iii) L ignores the inconsistent information in the construction of a summary value. Since there is plenty of evidence that recency of the interlocutor's inconsistent behavior plays a significant role in determining an attitude (Anderson, 1965), we will focus on the second type of reaction. Although the notion of 'recency' could be generalized to m latest elicited emotions, we simply refer to the very latest elicited emotion. Here, the update rule reads as follows.

$$\delta(Sit_n^{L,I}) = \delta^{\sigma}(Sit_{n-1}^{L,I}) \times \omega_h \mp \delta_w^{\overline{\sigma}}(Sit_n^{L,I}) \times \omega_r$$

The weights ω_h and ω_r denote the weights we apply to historical and recent information, respectively. ω_h and ω_r take values from the interval $[0, 1]$ and $\omega_h + \omega_r = 1$. A greater weight of recent information is reflected by using a greater value for ω_r. By way of example, let us assume that the agent likes its interlocutor with degree 3 and then gets angry at the interlocutor with intensity 5. The new value might be computed as $3 \times 0.25 - 5 \times 0.75$, resulting in a disliking value of 3.

The crucial question now is how the obtained (dis)liking value affects future interactions with the interlocutor. We consider two interpretations:

- *Momentary (dis)liking.* The new value is active for the current situation and then enters the summary record.
- *Essential (dis)liking.* The new value replaces the summary record.

Whereas momentary (dis)liking is a short-lived attitude change due to the elicitation of a high-intensity positive (negative) emotion, essential (dis)liking typically happens when the agent finds out something very positive (negative) about the interlocutor that is crucial for its model of the interlocutor.

Although the following observations are not reflected in our model, it is interesting to note that the way an agent deals with inconsistent information allows to make assumptions about its personality traits along the disagreeable–agreeable dimension. For instance, if the agent's attitude changes to essential disliking if made very angry once, it might be called unforgiving. Furthermore, a subtle interaction might exist between an agent's option for momentary or essential (dis)liking and the familiarity with the interlocutor. It can be argued that the most dramatic changes happen in recent evolving relationships, whereas agents familiar with each other rather experience momentary attitude changes.

3.5.2 Change of Social Variables

Social distance and power relationships have already been introduced in the context of emotion regulating factors. In this section, we will propose simple models that describe how social variables change. In modeling change of social distance, we are inspired by the work of Pautler and Quilici (1998) who investigate a special form of speech acts—called 'social perlocutions'— that may change the interlocutor's relationship with the agent. They argue that positive emotions elicited in the interlocutor contribute to improving the interlocutor's social relationship with the agent. Although they do not explicitly discuss relationships in terms of distance or familiarity, we belief that this interpretation is justifiable. The following update rule is based on the familiarity degree δ_F rather than the degree of social distance δ_D used before. However, they can be easily related by the definition $|\delta_F| := \delta_D - 5$ (where $|\delta|$ denotes the absolute value of δ). Note that only positive emotions elicited in L are considered.

$$\begin{aligned} \delta_F(Sit_0^{L,I}) &= 0 \text{ or pre-set to some value} \\ \delta_F(Sit_n^{L,I}) &= \delta_F(Sit_{n-1}^{L,I}) + \frac{\delta_w^+(Sit_n^{L,I})}{\pi} \end{aligned}$$

If a negative emotion is elicited, $\delta_F(Sit_n^{L,I}) = \delta_F(Sit_{n-1}^{L,I})$. π is a factor that determines how rapid an agent gets familiar with another agent. Unlike an agent's (dis)liking, familiarity increases monotonically, i.e., once agents are socially close, they cannot subsequently get unfamiliar. Currently, our notion of familiarity is based on the (severe) simplifying assumption that emotions are taken as the only familiarity changing factor. Cassell and Bickmore (2001), on the other hand, consider the variety and depth of topics covered by conversing agents.

Social power relationships are generally rather static, but there are well-defined situations where agents update their estimations. Consider a situation when an interlocutor says "You might not know me but I am your new manager" or when the interlocutor pulls a gun, then the agent will revise its hitherto assumed power status. Another form of revision occurs in situations where power can be assumed to be of less importance, e.g., when members of a social group (with different power levels) are engaged in a sportive activity. Here, system users can weight the impact of the power parameter, $\frac{\theta_P}{IMP_{\theta_P}}$, by choosing a larger value for IMP_{θ_P}.

4 System Implementation

Borrowing terminology from Perlin and Goldberg (1996), we distinguish two parts of our implemented system. The Animation Engine deals with a character's animated 'body' and associated capabilities such as speech recognition and synthetic speech output. The Behavior Engine is concerned with a character's 'mind' whose processes have already been discussed at length in this paper. In the following, we will first describe the functionality of an XML-style markup language (MPML) that greatly simplifies control of the Animation Engine. We start by explicating the language for scripted characters, where content authors are in full control of character behavior. After that, we will turn to the implementation of the Behavior Engine and its interface to the MPML script. Fig. 3 gives an overview of the implemented system.

4.1 A Markup Language for Character Control

We currently use the Microsoft Agent package (Microsoft, 1998) as our Animation Engine, which allows to embed animated characters into a web page based JavaScript interface. The package comes ready with controls for animating 2D cartoon-style characters, speech recognition and a Text-to-Speech (TTS) engine. Although character scripting with the Microsoft Agent package is straightforward, it soon gets cumbersome when the behavior of multiple characters has to be synchronized. In order to facilitate the process of scripting more complex scenarios, we developed an XML-style markup language called MPML

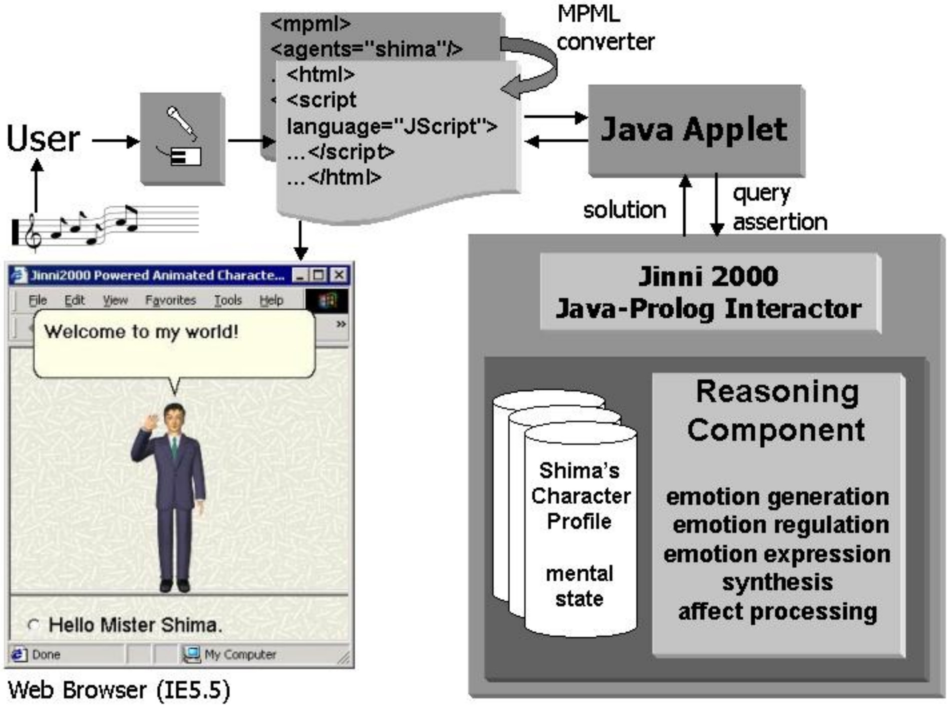

Figure 3: Implemented System Overview.

(Ishizuka et al., 2000; Descamps et al., 2001).[5] The primary responsibility of MPML is the control and synchronization of possibly multiple characters. Basic tagging schemes for a character's behavior are

- `<act/>` where a character performs a pre-defined animation sequence (including "alert", "blink", "decline", "explain", "greet", "sad", "suggest", etc.),
- `<move/>` where a character moves to a location given by its x and y coordinates,
- `<speak>...</speak>` where a character speaks a pre-defined sentence which is also displayed in a balloon,
- `<think>...</think>` where the sentence is only shown in a balloon next to the character, and
- `<listen>...</listen>` where the character is prepared to recognize a pre-defined sentence.

The overall organization of a (interactive) presentation script is defined by scenes that correspond to individual web pages. Within a scene, one or more characters may act sequentially, or multiple characters perform actions in parallel. Fig. 4 (top) shows the MPML script for a scene where the character "James" utters the sentence "Do you guys want to play Black Jack?", followed by the reply of character "Al" and then "Spaceboy".

[5]Strictly speaking, the version of MPML used here is not XML-based as it does not follow the usual paradigm of defining the form of the XML content script by eXtensible Stylesheet Language (XSL) authoring. Instead, the XML-style script is converted to a script that executable in a web browser (Internet Explorer 5.5 or higher), namely, JavaScript.

After that, Al speaks the sentence “Ready? You got enough coupons?” while Spaceboy is clapping with his hands.

MPML also provides the (empty) tag `<emotion/>` which can take an attribute such as the name-value pair `assign="james:happiness"` to modulate James’ synthetic speech output according to vocal effect on *happiness* described by Murray and Arnott (1995), and trigger an animation that makes James ‘smile’. If the effect should endure for a longer period, the `<mood>...</mood>` tagging scheme can be used.

In summary, MPML is a powerful and easy-to-use markup language that allows content authors to script rich web-based scenarios featuring animated characters. So far, we only considered characters with scripted behaviors, i.e., the author has full control over a character’s verbal and non-verbal behavior. This is sufficient for a wide range of characters that may populate a scenario. They may take the role of presenting certain information or play a minor role. However, for certain characters, e.g., those that interact with the user or play the heroine or hero in a game, we may want to achieve behaviors that are more adaptive to their environment. In the following, we will show how to relax the restriction to scripted behavior by interfacing MPML with SCREAM’s reasoning module that supports autonomous control of a character’s affective behavior.

4.2 Scripting a Character’s Mind

The Behavior Engine is implemented in Jinni 2000, a Java-based Prolog system (BinNet, 2000), and conceptually divided into two parts. The *Character Profile* declares the mental features of a character in the form of Prolog facts. Designated features are initialized with intensity values, some of which are dynamically updated as a result of the character’s interaction with its environment. The *Reasoning Component* contains a set of rules that encode a character’s mental processes. Although a separate Reasoning Component might be defined for each character, our current system assumes that all characters have access to the same reasoning capabilities. As shown in Fig. 3, Jinni 2000 communicates with a Java applet via Java-to-Jinni and Jinni-to-Java method calls.

Communication between MPML and the Java applet is realized by special tagging schemes (see Fig. 4, bottom, for an example). The `<execute/>` tag may call a Java method, e.g., to assert a communicative act of another agent to the character’s knowledge base. In order to retrieve the character’s reaction from the Behavior Engine, the `<consult>...</consult>` tagging scheme is used in conjunction with child tagging scheme `<test>...</test>`. Depending on the value of the `test` element, the character will perform a sequence of verbal and non-verbal behaviors.

5 Illustration

We will now illustrate how our systems works. As an interaction setting, we choose a casino scenario (shown in Fig. 5) where the user and other characters can play the “Black Jack” game. In our game scenario, the table seats a dealer and three players. The character “James” is in the role of the dealer. The first seat on the dealer’s left (the ‘First Base’) is occupied by the user who may interact with the game by uttering one of the sentences displayed in the lower frame window of the Internet Explorer window depicting the current scene. To the dealer’s right (the ‘Third Base’), the character “Al” and next to him “Space Boy” are in the role of other players. At the bottom left, the character “Genie” acts as the user’s advisor to play the game. Genie himself has an assistant, the

```
<!-- Example MPML script -->
<mpml>
...
  <scene id="introduction" agents="james,al,spaceboy">
    <seq>
      <speak agent="james">Do you guys want to play Black Jack?</speak>
      <speak agent="al">Sure.</speak>
      <speak agent="spaceboy">I will join, too.</speak>
      <par>
        <speak agent="al">Ready? You got enough coupons?</speak>
        <act agent="spaceboy" act="applause"/>
      </par>
    </seq>
  </scene>
...
</mpml>

<!-- MPML script illustrating interface with Jinni 2000 -->
<mpml>
...
  <consult target="[...].jamesApplet.askJinniResponseComAct('james','al','5')">
    <test value="resp25">
      <act agent="james" act="pleased"/>
      <speak agent="james">I am so happy to hear that.</speak>
    </test>
    <test value="resp26">
      <act agent="james" act="decline"/>
      <speak agent="james">We can talk about that another time.</speak>
    </test>
    ...
  </consult>
...
</mpml>
```

Figure 4: MPML example scripts.

character "Angel", who may direct the user's attention to important new events occurring in the game. Fig. 5 shows a situation where Genie practices Black Jack with the user by commenting Al's game. The dealer James has his own thoughts about the ongoing game, which is illustrated by the balloon next to him (the special shape of the balloon indicates that there is no synthetic speech output). The rules we use are fairly simple. If the dealer's hand is less or equal to 16, he must take a card ('hit'). Otherwise, if it is 17 or more, he must 'stand'. The participant closest to 21 wins the game. A character indicates that he wants to hit by saying "Hit" accompanied by a pointing gesture or nod "yes". The character indicates that he wishes to stand by nodding "no" (or some other appropriate gesture) while saying "Stand". For simplicity, we do not account for the possibility of 'splitting' in our Black Jack game.

In this paper, we solely focus on the affective reaction of the advisor Genie, who is in fact the only character who is driven by the SCREAM system. Genie advices the user to hit or stand, who may either follow or not follow the advice. The outcome of the game (whether the user wins or looses) is not determined by her or him following Genie's advice,

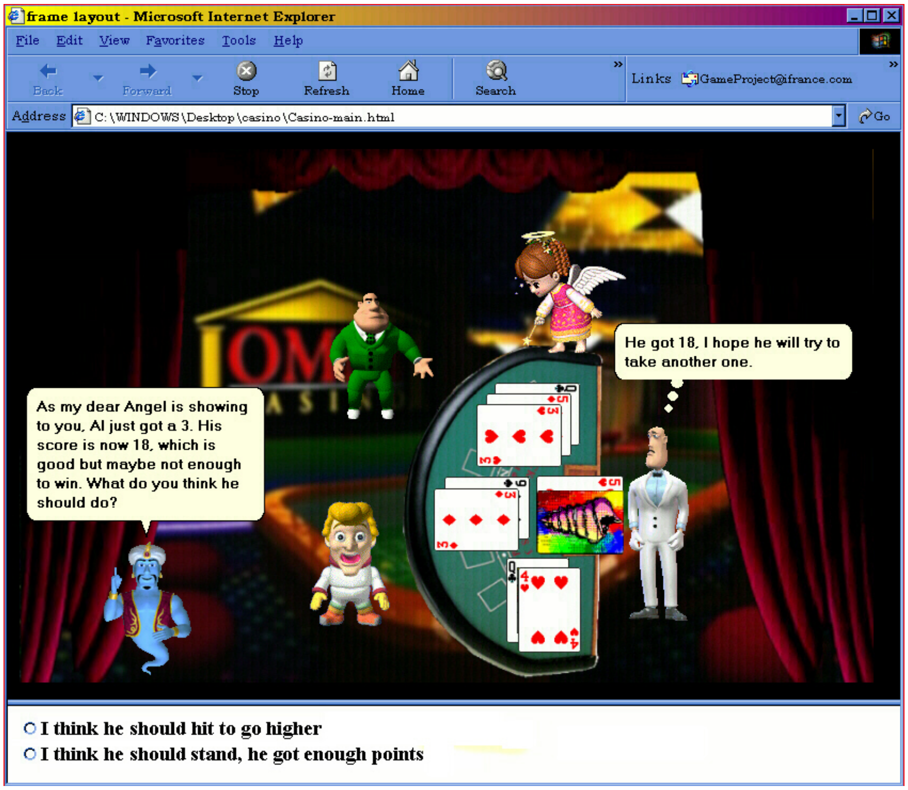

Figure 5: Casino Scenario.

i.e., she or he may loose even if following the advice. In that way, there is more room for variety in Genie's emotion expression.

In the following, we will watch the user playing five games of Black Jack. We intend to illustrate how Genie's mental makeup as well as the (affective) interaction history determine his behavior. In order to achieve more distinct reactions, we let the user either never or always follow Genie's advice. Furthermore, in order to be able to more easily track Genie's emotional reaction, we use a very sparse Character Profile, as shown in Fig. 6.[6] The influence of other parameters is explained and illustrated, e.g., in Prendinger and Ishizuka (2001a). The figure also contains the communicative act Genie receives in situation 0, together with its preconditions. Observe that the conditions include an 'interpretation' of the user's utterance in terms of its presumed emotional significance for Genie. This is strictly speaking not necessary and could by replaced by the previously mentioned stereotypical characterization of an interlocutor, e.g., by assuming that the user wants to win (with a certain default intensity). Moreover, note that we used a minor syntactical variant of the *com-act/5* relation discussed above, by adding the 'modalities' of the communicative act to its preconditions.

The dialogue between user and Genie where the user never follows Genie's advice is shown in Fig. 7. In the following, we will comment on the output.

Game One. Genie's winning emotional state is *distress* with intensity 4, because the

[6] In particular, we do not distinguish between different *types* of goals. Genie's goals "the user follows his advice" and "the user wins" will have a different impact on Genie if they are not satisfied, or even some interdependencies. A more accurate modeling would add the proposition *blameworthy(user_not_follows_advice)* leading to *reproach* and *angry at* emotional states.

```
% Personality traits
personality-type(advisor,agreeableness,3).      personality-type(advisor,extraversion,2).
% Social relationships (declared for all situations)
social-power(user,advisor,0,Sit).               social-distance(advisor,user,1,Sit).
% Attitude and (winning) emotional state in initial situation (-1)
likes(advisor,user,1,-1).                       joy(advisor,user,3,-1).
% Goals (declared for all situations)
wants(advisor,user_wins_game,1,Sit).            wants(advisor,user_follows_advice,4,Sit).
% The user's communicative act in situation 0
com-act(user,advisor,lost_game,0).
% Preconditions of the user's communicative act in situation 0
distress(user,advisor,user_looses_game,2,1).    linguistic_style_other(user,friendly,1,1).
holds(user_looses_game,1).                      holds(not_user_follows_advice,1).
```

Figure 6: Some features of the Genie's Character Profile.

user did not follow his advice. However, he displays *distress* with low intensity as his personality traits (friendly, extrovert) and the user's linguistic friendliness effect a decrease in the intensity of negative emotion expression. Precisely, since $\theta = 1$ and $\gamma = 2$, $\epsilon = 1$ $(= 4 - (1 + 2))$.

Game Two. Genie is sorry for the user with intensity 4, since positive (*sorry for*) emotions decay slowly and sum up, which leads to an increase in Genie's liking of the user. His personality traits let him express the emotion with even higher intensity.

Game Three. Genie gloats over the user's lost game, because at that point, the negative emotions dominate the positive ones as a consequence of the user's repeated refusal to follow Genie's advice. Hence Genie's attitude changes to slightly disliking the user which lets him to experience *joy* over the user's *distress* (*gloat* with intensity 5). Again, Genie's personality traits and the user's friendliness decrease the intensity of its external emotion to intensity 2.

Game Four. Here, Genie's winning emotional state is *bad mood* with intensity 5, slightly more that his *happy for* emotion (as the user wins the game this time). Here an overall, unspecific affective state (mood) is expressed with low intensity, rather than a specific emotion.

Game Five. Genie's dominant emotional state is *resent* with intensity 4, because he slightly dislikes the user and consequently is distressed that the user won by ignoring his advice. Change of social distance in dialogues of only five steps is minimal, and did not affect Genie's expression of emotions.

The dialogue where the user always follows Genie's advice is given in Fig. 8. Here, Genie's external emotions are more predictable, as he is shows *joy* or *happy for* emotions when the user wins and a *sorry for* emotion whenever the user looses. As the user follows Genie's advice, his attitude toward the user is increasingly positive, which elicits the *happy*

#	Speaker	Utterance	Expressed emotion (intensity)
1	Genie	You got only 16 now, so you should hit again.	
	User	*[stands, and looses]* Hmm, I lost.	
	Genie	*[sad face]* Oh. That was too little to stand.	*distress* (1)
2	Genie	You got 18 and should better stand.	
	User	*[hits, and looses]* Oh no, I lost.	
	Genie	*[sad face]* I am very sorry for you but in this case you better stand.	*sorry for* (5)
3	Genie	You got 14 and should hit again.	
	User	*[stands, and looses]* I lost.	
	Genie	*[smiling]* See! That's because you never follow my advice.	*gloat* (2)
4	Genie	Now you got 18 again. You'd better stand.	
	User	*[hits, and wins]* I did it!	
	Genie	*[frowning]* You are just lucky this time.	*bad mood* (2)
5	Genie	Now you have 19, that's too close to 21, so stand by all means.	
	User	*[hits, and wins]* I won!	
	Genie	*[frowning]* I cannot believe you are so lucky.	*resent* (1)

Figure 7: Dialogue for five consecutive games where the user never follows Genie's advice. The table does not show the user's interaction with the dealer.

for emotion in the third game (rather than *joy*). Furthermore, Genie's personality traits (friendly and extrovert) ensure that he expresses positive emotions with high intensity.

We will conclude this section with two remarks concerning Genie's affective behavior. *First*, calling Genie an 'advisor' might be slightly misleading in the sense that we do not provide him with any dialogue strategies typically found in pedagogical agents (Johnson et al., 2000; Conati, 2001). Genie's reactions are determined by the character profile only. In order to turn Genie into a tutor for the Black Jack game, we might link his affective reasoning model to a decision network (as discussed above) where his (re)actions are explicitly dependent on their utility for improving the user's game playing.

Second, we only described Genie's reaction for the 'extreme' cases where the user never or always follows his advice. Since the user may either follow or not follow Genie's advice at each of the five decision points, possible interactions can be seen as traversals through a binary tree of depth five. Hence, a total of 32 ($= 2^5$) interaction patterns are obtained. In the remaining cases, Genie's reactions are conform at the beginning steps of the interaction, whereas they get more diverse toward the end. The overwhelming portion of Genie's reactions are positive, which is consistent with his personality traits. In fact, the crucial feature is Genie's attitude toward the user, which lets him either be sorry for or gloat over the user's lost game, and be happy for or resent if the user wins a game.

#	Speaker	Utterance	Expressed emotion (intensity)
1	Genie	You got only 16 now, so you should hit again.	
	User	*[hits, and wins]* I won!	
	Genie	*[smiling]* Great! You did it!	*joy* (5)
2	Genie	You got 18 and should better stand.	
	User	*[stands, and wins]* I won!	
	Genie	*[smiling]* Good job!	*joy* (5)
3	Genie	You got 14 and should hit again.	
	User	*[hits, and wins]* I did it!	
	Genie	*[congratulate]* I am so happy for you. You got a lucky day.	*happy for* (5)
4	Genie	Now you got 18 again. You'd better stand.	
	User	*[stands, and looses]* Oh, I lost.	
	Genie	*[sad face]* I am so sorry for you.	*sorry for* (5)
5	Genie	Now you have 19, that's too close to 21, so stand by all means.	
	User	*[stands, and looses]* Hmm, that was unlucky.	
	Genie	*[sad face]* I am so sorry that you lost.	*sorry for* (5)

Figure 8: Dialogue for five consecutive games where the user always follows Genie's advice. The table does not show the user's interaction with the dealer.

6 Discussion and Conclusion

In this paper, we discuss models and tools for scripting and coordinating affective interactions with and among animated characters. While MPML is a powerful tool for controlling and coordinating the visual behavior of characters (their 'body'), the SCREAM system constitutes a practical technology for scripting the mental processes underlying a character's affective behavior (its 'mind'). Its flexibility derives from the granularity feature, i.e., the author may decide on the level of detail at which the character is scripted. If the author wishes to introduce many levels of indirection to the agent's behavior, she or he may define all of the available parameters and also control the influence of each parameter on emotional states by editing the combination functions. In certain settings, however, only a subset of the parameters might be of interest, e.g., when the author wants to script a (interactive) presentation agent that is only driven by goals and personality. The system will manage the elicited emotions and produce an output that reflects the provided influences. Authors also have the option to script some of the character's reactions directly, without activating the agent's mind.

A major concern about the SCREAM system is that it assumes a rich repertoire of 'canned' affective verbal responses that reflect both the expressed emotion and its intensity. Petta and Trappl (1997, p. 210–214) address this as a problem of 'shallow' (or 'top-down') approaches to characters with synthetic personality, and seem to favor 'deep' ('generative', 'bottom-up') approaches that (they argue) allow for the dynamic and autonomous extension of a character's available behaviors. However, given our explicit focus on *scripting* interactive affective behavior, we have to find other ways to alleviate the effort of preparing a huge number of character reactions. Obvious simplifications consist in collapsing emotion types or intensity levels. Since emotions are divided into positive and negative ones, reactions can be abstracted to 'good mood' and a 'bad mood' responses. Furthermore, intensity levels can be decreased by merging them, e.g., as 'neutral', 'low intensity', and 'high intensity'. Rather than abstracting away the carefully designed accuracy of the system to provide a character with an appropriate affective response, authors might prefer some automatization of answer generation. Walker et al. (1997) propose algorithms for Linguistic Style Improvisation (LSI) based on the social variables discussed in Brown and Levinson (1987) that might serve as a starting point for automatically generating surface forms that reflecting emotion and intensity. Specifically, LSI strategies determine semantic content, syntactic form and acoustic realization of a speech act, qualified by the social context. Application of LSI strategies supports social interactions that allow agents to maintain 'public face' (their desire for autonomy and approval). If speaker and hearer have equal social rank, 'direct' strategies can be applied (e.g., "I told you not to 'hit' when the dealer's hand is 19!"). On the other extreme, when the rank distance is large, 'off record' strategies are chosen (e.g., "Often people refrain from 'hitting' when the dealer's hand is 19.").

Another issue of the current system architecture is that it is very rigid regarding the content and timing of a user's dialogue contribution. Borrowing the terminology from Allen et al. (2001), we deal with a fixed-initiative finite state script where users are prompted to utter their dialogue contribution that is drawn from a very limited choice of possible utterances. User-agent communication would certainly be more natural if the user could interrupt the character's speech and take initiative at any time during the conversation. We hope to address this problem in our future work.

For the time being, the main focus of our research is the design of life-like characters that are believable in their affective reactions and controllable in web-based environments. With the development of the SCREAM and MPML systems, we hope to have provided tools that greatly facilitate this endeavor.

Acknowledgments

This research is supported by the Research Grant (1999-2003) for the Future Program ("Mirai Kaitaku") from the Japanese Society for the Promotion of Science (JSPS). The first author would like to express his special thanks to Fiorella de Rosis for highly illuminating discussions on emotional agents.

References

Alchourrón, C. E., P. Gärdenfors, and D. Makinson (1985). On the logic of theory change: partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2): 510–530.

Allen, J. F., D. K. Byron, M. Dzikovska, G. Ferguson, L. Galescu, and A. Stent (2001). Toward conversational human-computer interaction. *AI Magazine*, 22(4):27–37.

S. Allen, S. (1999). Control states and motivated agency. In E. André, editor, *Proceedings* *of the i3 Spring Days'99 Workshop on Behavior Planning for Life-like Characters and Avatars*, pages 43–61.

Anderson, N. H. (1965). Primacy effects in impression formation using a generalized order effect paradigm. *Journal of Personality and Social Psychology*, 2:1–9.

André, E., M. Klesen, P. Gebhard, S. Allen, and T. Rist (1999). Integrating models of personality and emotions into lifelike characters. In *Proceedings International Workshop on Affect in Interactions - Towards a New Generation of Interfaces*, pages 136–149.

André, E., T. Rist, S. van Mulken, M. Klesen, and S. Baldes (2000). The automated design of believable dialogue for animated presentation teams. In J. Cassell, S. Prevost, J. Sullivan, and E. Churchill, editors, *Embodied Conversational Agents*, pages 220–255. The MIT Press.

Arafa, Y., and A. Mamdani (1999). Virtual personal service assistants: Real-time characters with artificial hearts. In *Proceedings IEEE 5th International Conference on Systems, Man and Cybernetics (SMC-99)*, pages 762–767.

Badler, N. I., J. A. Rama Bindiganavale, W. Schuler, L. Zhao, and M. Palmer (2000). Parameterized action representation for virtual human agents. In J. Cassell, S. Prevost, J. Sullivan, and E. Churchill, editors, *Embodied Conversational Agents*, pages 256–284. The MIT Press.

Ball, G., and J. Breese (2000). Emotion and personality in a conversational agent. In J. Cassell, S. Prevost, J. Sullivan, and E. Churchill, editors, *Embodied Conversational Agents*, pages 189–219. The MIT Press.

Barwise, J., and J. Perry (1983). *Situations and Attitudes*. MIT Press, Cambridge, MA.

BinNet Corp. (2000). *Jinni 2000: A high performance Java based Prolog for agent scripting, client-server and internet programming*. BinNet Corp. URL: `www.binnetcorp.com`.

Blumberg, B. M. (1996). *Old Tricks, New Dogs: Ethology and Interactive Creatures*. PhD thesis, Massachusetts Institute of Technology.

Brown, P., and S. C. Levinson (1987). *Politeness. Some Universals in Language Usage*. Cambridge University Press.

Cassell, J., and T. Bickmore (2001). Negotiated collusion: Modeling social language and its relationship effects in intelligent agents. User Modeling and Adaptive Interfaces. In Press.

Cassell, J., T. Bickmore, L. Campbell, H. Vilhjálmsson, and H. Yan (2000). Human conversation as a system framework: Designing embodied conversational agents. In J. Cassell, S. Prevost, J. Sullivan, and E. Churchill, editors, *Embodied Conversational Agents*, pages 29–63. The MIT Press.

Cassell, J., H. Vilhjálmsson, and T. Bickmore (2001). BEAT: the Behavior Expression Animation Toolkit. In *Proceedings of SIGGRAPH-01*, pages 477–486.

Conati, C. (2001). Modeling students' emotions to improve learning with educational games. In *Proceedings of 2001 AAAI Fall Symposium in Emotional and Intelligent II: The Tangled Knot of Social Cognition*, pages 31–36. AAAI Press, Technical Report FS-01-02.

de Carolis, B., C. Pelachaud, I. Poggi, and F. de Rosis (2001). Behavior planning for a reflexive agent. In *Proceedings 17th International Conference on Artificial Intelligence (IJCAI-01)*.

de Rosis, F., B. de Carolis, and S. Pizzulito (1999). Software documentation with animated agents. In *Proceedings 5th ERCIM Workshop on User Interfaces For All*.

Descamps, S., H. Prendinger, and M. Ishizuka (2001). A multimodal presentation mark-up language for enhanced affective presentation. In *Proceedings International Conference on Intelligent Multimedia and Distant Education (ICIMADE-01), Advances in Educational Technologies: Multimedia, WWW and Distance Education*, pages 9–16.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3–4): 169–200.

Ekman, P., and W. V. Friesen (1969). The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica*, 1:49–98.

Elliott, C. (1992). *The Affective Reasoner. A process model of emotions in a multi-agent system.* PhD thesis, Northwestern University. Institute for the Learning Sciences.

Elliott, C., and A. Ortony (1992). Point of view: Modeling the emotions of others. In *Proceedings 14th Annual Conference of the Cognitive Science Society*, pages 809–814.

Extempo Systems Inc., 2001. URL: `www.extempo.com`.

Gmytrasiewicz, P. J., and C. Lisetti (2000). Using decision theory to formalize emotions for multi-agent systems. In *Second ICMAS-2000 Workshop on Game Theoretic and Decision Theoretic Agents*.

Gratch, J. (2000). Émile: Marshalling passions in training and education. In *Proceedings 4th International Conference on Autonomous Agents (Agents'2000)*, pages 325–332.

Gross, J. J. (1998). The emerging field of emotion regulation: An integrative review. *Review of General Psychology*, 2(3):271–299.

Hayes-Roth, B. (2001). Adaptive learning guides. In *Proceedings of the IASTED Conference on Computers and Advanced Technology in Education*.

Huber, M. J. (1999). JAM: A BDI-theoretic mobile agent architecture. In *Proceedings 3rd International Conference on Autonomous Agents (Agents-99)*, pages 236–243.

Ishizuka, M., T. Tsutsui, S. Saeyor, H. Dohi, Y. Zong, and H. Prendinger (2000). MPML: A multimodal presentation markup language with character control functions. In *Proceedings Agents'2000 Workshop on Achieving Human-like Behavior in Interactive Animated Agents*, pages 50–54.

Johnson, W. L., J. W. Rickel, and J. C. Lester (2000). Animated pedagogical agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education*, 11:47–78.

Lewis, D. (1969). *Convention. A Philosophical Study.* Harvard University Press.

Mateas, M., and A. Stern (2000). Towards integrating plot and character for interactive drama. In *Proceedings AAAI Fall Symposium on Socially Intelligent Agents: The Human in the Loop.*

McBreen, H., P. Shade, M. Jack, and P. Wyard (2000). Experimental assessment of the effectiveness of synthetic personae for multi-modal e-retail applications. In *Proceedings 4th International Conference on Autonomous Agents (Agents'2000)*, pages 39–45.

Microsoft (1998). *Developing for Microsoft Agent.* Microsoft Press.

Moffat, D. (1997). Personality parameters and programs. In R. Trappl and P. Petta, editors, *Creating Personalities for Synthetic Actors*, pages 120–165. Springer.

Moulin, B. (1998). The social dimension of interactions in multiagent systems. In W. Wobcke, M. Pagnucco, and C. Zhang, editors, *Agents and Multi-Agent Systems, Formalisms, Methodologies and Applications*, pages 109–122. Springer Verlag.

Murray, I. R., and J. L. Arnott (1995). Implementation and testing of a system for producing emotion-by-rule in synthetic speech. *Speech Communication*, 16:369–390.

O'Rorke, P., and A. Ortony (1994). Explaining emotions. *Cognitive Science*, 18:283–323.

Ortony, A. (1991). Value and emotion. In W. Kessen, A. Ortony, and F. Craik, editors, *Memories, thoughts, and emotions: Essays in the honor of George Mandler*, pages 337–353. Hillsdale, NJ: Erlbaum.

Ortony, A. (2001). On making believable emotional agents believable. In R. Trappl, P. Petta, and S. Payr, editors, *Emotions in Humans and Artifacts.* The MIT Press.

Ortony, A., G. L. Clore, and A. Collins (1988). *The Cognitive Structure of Emotions.* Cambridge University Press.

Paiva, A., I. Machado, and R. Prada (2001). Heroes, villains, magicians,...: Dramatis personae in a virtual story creation environment. In *Proceedings International Conference on Intelligent User Interfaces (IUI-2001)*, pages 129–136.

Pautler, D., and A. Quilici (1998). A computational model of social perlocutions. In *Proceedings of COLING-ACL-98.*

Perlin, K., and A. Goldberg (1996). Improv: A system for scripting interactive actors in virtual worlds. *Computer Graphics*, 30:205–216.

Petta, P., and R. Trappl (1997). Personalities for synthetic actors: Current issues and some perspectives. In R. Trappl and P. Petta, editors, *Creating Personalities for Synthetic Actors*, pages 209–218. Springer.

Picard, R. W. (1997). *Affective Computing.* The MIT Press.

Poole, D., A. Mackworth, and R. Goebel (1998). *Computational Intelligence. A Logical Approach.* Oxford University Press.

Prendinger, H., and M. Ishizuka (2001a). Agents that talk back (sometimes): Filter programs for affective communication. In *Proceedings 2nd Workshop on Attitude, Personality and Emotions in User-Adapted Interaction, in conjunction with User Modeling 2001.*

Prendinger, H., and M. Ishizuka (2001b). Let's talk! Socially intelligent agents for language conversation training. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, 31(5):465–471.

Prendinger, H., and M. Ishizuka (2001c). Social role awareness in animated agents. In *Proceedings 5th International Conference on Autonomous Agents (Agents-01)*, pages 270–277.

Reilly, W. S. N. (1996). *Believable Social and Emotional Agents.* PhD thesis, Carnegie Mellon University. CMU-CS-96-138.

Rich, E. (1979). User modeling via stereotypes. *Cognitive Science*, 3:329–354.

Ringate, T. (2001). *AIML Reference Manual.* A.L.I.C.E AI Foundation. URL: `alicebot.org`.

Rousseau, D., and B. Hayes-Roth (1998). A social-psychological model for synthetic actors. In *Proceedings 2nd International Conference on Autonomous Agents (Agents-98)*, pages 165–172.

Russell, S. J., and P. Norvig (1995). *Artificial Intelligence. A Modern Approach.* Prentice Hall, Inc., Upper Saddle River, New Jersey.

Verbot (1998). Virtual Personalities Inc., Verbot Sylvie Ver. 3.04.

Walker, M. A., J. E. Cahn, and S. J. Whittacker (1997). Improvising linguistic style: Social and affective bases for agent personality. In *Proceedings First International Conference on Autonomous Agents (Agents'97).*