SIMULATION OF NONVERBAL SOCIAL INTERACTION AND SMALL GROUPS DYNAMICS IN VIRTUAL ENVIRONMENTS

THÈSE N^{O} 2933 (2004)

PRÉSENTÉE À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS

Institut des systèmes informatiques et multimédias

SECTION D'INFORMATIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Anthony GUYE-VUILLÈME

Maîtrise ès sciences sociales de l'Université de Lausanne
de nationalité suisse et originaire des Bayards (NE)

acceptée sur proposition du jury:

Prof. D. Thalmann, directeur de thèse
Prof. P. Dillenbourg, rapporteur
Dr T. Rist, rapporteur
Dr D. Schneider, rapporteur

Lausanne, EPFL
2004

Remerciements

Pour le précieux soutien qu'elles m'ont apporté, chacune à leur manière, pendant ces enrichissantes années d'études, de travail et parfois aussi de doutes, je souhaite remercier chaleureusement les personnes suivantes:

Mon directeur de thèse, le Professeur Daniel Thalmann, qui m'a accordé sa confiance malgré un profil académique peu classique dans le contexte de l'EPFL, et qui m'a permis de réaliser cette recherche dans les meilleures conditions.

Les rapporteurs de cette thèse, le Professeur Pierre Dillenbourg, le Docteur Thomas Rist et le Docteur Daniel Schneider, pour la grande pertinence de leurs commentaires et suggestions.

Le jury dans son ensemble, pour l'inattendu honneur qu'il m'a fait en proposant ce travail pour le prix EPFL de doctorats.

Le Docteur Tolga Çapin, qui m'a fait croire en ma capacité à réaliser une telle recherche et m'a amicalement mis le pied à l'étrier.

Georges Dumoulin et Akram Hajjaoui, qui m'ont permis, par la grande compréhension dont ils ont fait preuve, d'aménager au mieux mon emploi du temps pour la réalisation de ce travail.

Fabrice Vergnenegre, le Docteur Frédéric Vexo, Mireille Clavien, Michal Ponder, Sébastien Schertenleib, Bruno Herbelin, Olivier Renault, Josiane Bottarelli et Josiane Gisclon, pour l'aide précieuse qu'ils m'ont apportée chacun dans son domaine.

Etienne de Sevin, Pascal Glardon, Raquel Urtasun, Benoît Le Callennec, Luca Vacchetti, Stéphanie Noverraz, Thierry Michellod, Branislav Ulicny, et les autres assistants du VRLab et du CVLab pour les bons moments passés ensemble.

Mes parents, Rose-Marie et Jean-Frédéric Guye-Vuillème, pour leur aide et leurs constants encouragements.

Last but not least, Beatriz Dias, interlocutrice aux compétences multiples et soutien de tous les instants, sans qui rien n'aurait été possible.

Abstract

How can the behaviour of humans who interact with other humans be simulated in virtual environments? This thesis investigates the issue by proposing a number of dedicated models, computer languages, software architectures, and specifications of computational components. It relies on a large knowledge base from the social sciences, which offers concepts, descriptions, and classifications that guided the research process. The simulation of nonverbal social interaction and group dynamics in virtual environments can be divided in two main research problems: (1) an action selection problem, where autonomous agents must be made capable of deciding when, with whom, and how they interact according to individual characteristics of themselves and others; and (2) a behavioural animation problem, where, on the basis of the selected interaction, 3D characters must realistically behave in their virtual environment and communicate nonverbally with others by automatically triggering appropriate actions such as facial expressions, gestures, and postural shifts.

In order to introduce the problem of action selection in social environments, a high-level architecture for social agents, based on the sociological concepts of role, norm, and value, is first discussed. A model of action selection for members of small groups, based on proactive and reactive motivational components, is then presented. This model relies on a new tag-based language called Social Identity Markup Language (SIML), allowing the rich specification of agents' social identities and relationships. A complementary model controls the simulation of interpersonal relationship development within small groups. The interactions of these two models create a complex system exhibiting emergent properties for the generation of meaningful sequences of social interactions in the temporal dimension.

To address the issues related to the visualization of nonverbal interactions, results are presented of an evaluation experiment aimed at identifying the application requirements through an analysis of how real people interact nonverbally in virtual environments. Based on these results, a number of components for MPEG-4 body animation, AML - a tag-based language for the seamless integration and synchronization of facial animation, body animation, and speech - and a high-level interaction visualization service for the VHD++ platform are described. This service simulates the proxemic and kinesic aspects of nonverbal social interactions, and comprises such functionalities as parametric postures, adapters and observation behaviours, the social avoidance of collisions, intelligent approach behaviours, and the calculation of suitable interaction distances and angles.

Résumé

Comment simuler dans les environnements virtuels le comportement d'humains en interaction avec d'autres humains ? Cette thèse contribue à l'avancement de la recherche dans ce domaine en proposant différents modèles comportementaux, languages, architectures et composants logiciels. Les contributions présentées sont fortement basées sur l'apport des sciences sociales, qui, par les concepts, descriptions et classifications qu'elles proposent, ont guidé le processus de recherche et de développement. La simulation des interactions sociales non-verbales et de la dynamique des groupes restreints dans les environnements virtuels peut être divisée en deux problèmes principaux: (1) un problème de sélection d'action, qui consiste à permettre à des agent autonomes de décider quand, avec qui, et comment ils interagissent, en fonction de leurs caractéristiques et de celles des autres agents, (2) un problème d'animation comportementale, qui consiste à représenter visuellement de manière réaliste, à partir des types d'interaction sélectionnés, les comportements des personnages virtuels en générant un flux d'animations faciales et corporelles appropriées.

Afin d'introduire le problème de la sélection d'action dans un contexte social, une architecture de haut niveau pour agents sociaux, basée sur les concepts sociologiques de rôle, norme et valeur, est tout d'abord discutée. Un modèle de sélection d'action applicable aux membres de groupes restreints, comprenant des composants aussi bien réactifs que proactifs, est ensuite présenté. Ce modèle est basé sur SIML, un language de type XML developpé pour permettre la spécification aisée de différentes identités sociales. Un second modèle simule l'évolution des relations interpersonnelles au sein des groupes restreints. Les interactions de ces deux modèles créent un système complexe ayant des propriétés émergentes, permettant la génération de séquences d'interactions réalistes dans le temps.

Pour déterminer les besoins en termes de fonctionnalités liés à la visualisation des interactions non-verbales, les résultats d'une évaluation qualitative de la manière dont des sujets réels agissent entre eux dans le contexte d'un environnement virtuel sont tout d'abord présentés. Un certain nombre de composants pour le standard d'animation MPEG-4, AML - un language permettant l'intégration et la synchronisation d'animations faciales, corporelles et de la synthèse de parole, ainsi qu'une service de visualisation d'interactions non-verbales pour la plateforme logicielle VHD++ sont ensuite décrits. Il est expliqué comment ce service permet la simulation des aspects kinésiques et proxémiques des interactions sociales, comprenant entre autre la génération automatique de postures et de comportements d'observation, l'évitement social de collision, des trajectoires d'approche intelligentes, et le positionnement adéquat des participants aux interactions en termes de distance et d'orientation.

Table of Contents

Chapter 1 • Introduction

1.1 Motivation and Objectives

Recent advances in computer graphics have allowed the modelling and visualization of physical reality to reach an extremely high degree of realism. Similarly, many of the physical laws that control the inanimate objects of our world are now well understood, and it is possible to simulate accurately the forces that determine their behaviour.

In contrast, living creatures introduce a new level of complexity in the building of a realistic virtual reality. Artificial Life is a young and dynamic field whose goal is to address the issue of simulating living systems. Research in this field can successfully build on the momentum generated by the biological sciences. However, there is a research problem that, despite recently increased interests and efforts, still appears extremely challenging: the simulation of human behaviour, and in particular, the simulation of humans' behaviour with other humans.

We consider that at least three factors explain the encountered difficulties:

1. **The complexity of the phenomenon to be simulated.** Human beings are biological creatures that are the result of a long evolutionary process, but although every individual is subjected to similar physical needs, their behaviour varies greatly according to their historical and cultural context. In addition, human beings experience complex emotions, are strongly interrelated and interdependent, and have a reflexive view of their own existence and behaviours.

2. **The weakness in our understanding of the processes that guide human action.** This point is primarily related to the complexity of the research topic; however it also appears that the humanities are frequently divided beyond dialogue and are the scene of many unfruitful internal quarrels. We include the social sciences in this assessment. This situation has prevented the development of integrated theoretical frameworks that account for all of the dimensions of human behaviour, and has favoured work on isolated case studies with limited heuristic value.

3. **The division between computer science and social sciences.** Although there are some early signs that this situation could be changing, these two fields have traditionally evolved completely separately. Computer scientists generally ignore the considerable body of knowledge available from the social sciences, in spite of the mentioned problems. When confronted with the simulation of some aspects of human behaviour, they have a tendency to start the research from scratch, to make purely ad-hoc choices, or to select very questionable theories that match their initial beliefs. On the other hand, social scientists too often choose to disregard the great explanatory power provided by numerical models and computer simulation.

The objective of our work is to explore the 3D simulation of nonverbal social interaction between humans in a face-to-face and group context. As such, it is subject to the limitations just described; however, we choose to ground our contribution in the concepts and findings of the social sciences, and hope that it can contribute to the establishment of a fruitful dialogue between the two fields.

The research problem of simulating nonverbal social interaction in virtual environments has several dimensions: the participants must be visually represented in 3D space; the individual's and group's social characteristics must be correctly modelled; and, on the basis of these characteristics, adequate face and body actions must be triggered and displayed in the virtual environment. Some examples of individual characteristics influencing nonverbal interactions are the participants' gender, age, personality and relationships with others. In addition to the purely computer graphics and animation aspects, the question of "when, with whom, and how does an interaction take place" is therefore at the heart of our topic.

We would like to stress that this complex problem can only be solved partially. Our work does not attempt to cover all possible situations, which vary almost infinitely according to the social environment (work, living or play), the physical environment (seated or standing), the participants' identity (culture, gender, etc.), or the group properties (size, history, etc.). Although some task-oriented interactions are included in our simulations, we are more interested in the small "social" groups such as people in pubs, "street-corners" groups, sport and game clubs, etc., which do not have explicitly defined specific rules of conduct. It is an important part of the work to propose relevant criteria to characterize individuals and groups; however, the list is obviously limited compared to the diversity of the real world. Standing encounters are also favoured, allowing for the expression of richer nonverbal processes compared to seated interactions, where the participants are constrained in their position choices, gestures, etc.

The proper simulation of nonverbal social interaction is valuable in several application domains:

- **Human–Computer Interaction (HCI).** As discussed in Chapter 3, the quality of the user's experience can be dramatically increased by using Embodied Conversational Agents, capable of realistic interactions, as an interface to the computer.

- **Production of content for the entertainment industry.** There is a growing use of virtual characters in movies, television and video games, and it would be very advantageous to be able to populate the scenes with synthetic actors able to interact with each other automatically and realistically. In a recent paper discussing the future of the gaming industry, Stephen Cass (2002) acknowledged this trend, writing "The next big challenge for game AI may be getting a game's cast of characters better at learning and social interaction".

- **Nonverbal communication and applied social psychology.** The simulation of nonverbal interaction can be used to validate, invalidate and improve the available models by asking subjects to evaluate the generated visuals. It can also be useful in cognitive therapies, in the simulation of emergency situations, etc.

In contrast with the artistic approach, where designers manually model the attitudes of the interacting virtual characters according to their own perception, the aim of this thesis is to provide a system based on verified data that allows for the automatic generation and systematic exploration of the nonverbal behaviours associated to a given interactional context.

1.2 Definitions

Several concepts appearing in the title of this document are worth considering more closely, since they are frequently used in order to express different ideas or because our usage of the term needs to be clarified.

1.2.1 Social Behaviour and Social Interaction

The term "social" is central in our work. Unfortunately, there is no general agreement on what it refers to. In the computer graphics field, the implicit definition of social behaviour is often restricted to crowd and group activities, for example in the work by Bouvier (1996). Agent researchers usually link the concept with a communication task and define social ability as the capability of interacting with other agents or humans (Wooldridge 1995). In ethology, social behaviour is related to the existence of casts and division of labour (Wcislo 2000).

These definitions refer to important characteristics, but the concept has a broader meaning in sociology. All human activity is defined as "social" since it is purely the product of a social-cultural context: "Men together produce a human environment, with the totality of its socio-cultural and psychological formations [..] As soon as one deserves phenomena that are specifically human, one enters the realm of the social. Man's specific humanity and his sociality are inextricably intertwined. Homo sapiens is always, and in the same measure, homo socius" (Berger 1966, p. 51). In contrast with the preceding views, the behaviour of temporarily isolated humans is here still social. It is not the fact of interacting with other people itself that makes for a social being, but the internalisation, through a situated process of socialization, of a number of fundamental beliefs about the world and about his/her position in society (Berger 1966; Bourdieu 1990).

However, this process of socialization takes place largely via interpersonal interactions. Symbolic interactionism, one of the major sociological schools, is dedicated to the study of the processes constructing, through social interaction, a shared definition of reality (Blumer 1969). In a different approach, functionalists have also underlined the importance of social interaction, for example as a means to influence other people (Bales 1951). More generally, it can be said that social interaction consists in the structural coupling of several organisms, involving some degrees of mutual perturbation (Maturana & Varela 1987).

1.2.2 Nonverbal Communication and Nonverbal Behaviour

Nonverbal activity is an important dimension of social interaction. Psychological studies have concluded that more than 65 percent of the information exchanged during a face-to-face interaction is expressed through nonverbal means (Argyle 1988). Body postures and movements, facial expressions, tones of voice, etc., augment spoken messages and give substance to social interaction.

The use of the body in interpersonal communication has been studied in psychology under the name of "Nonverbal Communication". The definition of this field is based on an exclusion: one defines nonverbal communication as the use of the whole set of means by which human beings communicate except for the human linguistic system and its derivatives (writings, sign language, etc.). However, in many cases, nonverbal communication is not used by itself but jointly with verbal communication. For instance, nonverbal signals can be used to underline the importance of a given sentence or to indicate that the speech is over. In contrast, there is a type of signal which is still independent from language: the affective expression. Several researchers have demonstrated that nonverbal signals do not need any verbal expression in the task of communicating emotional messages, and that they are able to express in a powerful way feelings that would be very difficult to express using the linguistic system (Corraze 1980).

The intentionality of nonverbal actions, the fact that someone intends to send a specific message or not, is an important characteristic that must be taken into account (Kendon 1981). We use the expression "nonverbal communication" when it is the goal of a sender of a message that it be understood by the recipient, and "nonverbal behaviour" in order to describe both intended/conscious and unintended/unconscious nonverbal activity.

1.2.3 Small Groups and Group Dynamics

The interest for small groups in sociology appears with the founding fathers, in particular Simmel and Cooley who strongly contributed to the modern definition of the concept. Simmel is mainly interested in the size of groups as a determinant of the type of social behaviours taking place in this context (Hare, Borgatta & Bales 1966). He explains that dyads should not be considered as groups: since the secession of either participants would result in the destruction of every relationship, dyads do not constitute a social structure *per se*. On the contrary, groups of three and more people allow their members to have indirect relationships with one another, and can maintain a given organization in spite of personnel changes. Simmel also identifies several qualities of smaller groups that inevitably disappear when the groups grow larger, like the fact that every member has a direct relationship with every other. Cooley adds the dimension of intimacy to the definition, and underlines the face-to-face interaction of the members of small groups as a fundamental criterion (Hare, Borgatta & Bales 1966). Following these authors, we consider in our work that small groups are composed of 3 to 20 members which have a direct relationship to each other and interact in face-to-face situations.

The expression "group dynamics" is first introduced by Kurt Lewin (1948). His goal is to emphasize the complex and changing nature of the behaviours taking place in human groups. Lewin's famous formula $B=f(P,E)$ generalizes his view that the behaviour of group members (B) is a function (f) of their personal traits (P) and of their environment (E), which includes the other group members with their specific individual characteristics and relationships. For him, group members are interdependent to a significant degree because they influence each other through social interaction, and thus, cannot be studied in isolation.

More recently, such researchers as Maturana and Varela (1987), Luhmann (1995), and Hejl (1984), had a strong impact on the conceptualisation of social systems, including organizations and small groups. With the concept of "autopoiesis", they emphasize the self-producing and self-maintaining properties of social systems resulting from the interaction of their constitutive elements. Maturana (1980), for example, argues that social systems are constituted by the interactivity of their participants, which is realized in linguistic domains, and vary according to changes in interaction frequency, members' connectivity, etc. In return, the social system influences individual participants through affordances for and regularities in their interactivity. This new way of approaching social processes is today having a revitalizing effect on the study of small groups, a field whose existence appeared to be threatened in the early 70s (Burke 2003).

We consider that small groups and their dynamics constitute a framework for the generation of interactions and, at the same time, that they are themselves constituted by these interactions. The described characteristics of small groups show that a specific approach is required compared to the previous work on both individual agents and crowd modelling.

1.3 Literature Review

Since our work is relevant to various research domains, we classify the previous contributions that we wish to reference in several categories: Artificial Intelligence and Autonomous Agents, Artificial Life and Action selection, Social Simulation and Social Agents, Behavioural Animation and Embodied Agents. We describe classic works but also include a number of interdisciplinary contributions which could have been placed in more that one category.

1.3.1 Artificial Intelligence and Autonomous Agents

One of the major contributions of the Artificial Intelligence field is certainly the introduction of the "production system" planning model, used in well-known applications like GPS (Newell & Simon 1972) and Soar (Laird, Newell & Rosenbloom 1987). This approach is strongly related to our work, since it is based on the study of human cognitive processes. The main components of a production system are a database of production rules, at least one declarative memory segment for storing initial and generated beliefs, and a controller. When used for action selection, the rules are plans that allow for the satisfaction of some goals either directly or through the production of sub-goals. Since pattern-matching techniques are included in the controller, generic rules, which are able to deal with different kinds of entities, can be provided. The system can be used in forward-chaining mode, i.e. starting from its database of facts, the controller continually checks the rules until a goal is achieved, or in backward-chaining mode, i.e., starting from a goal hypothesis, it recursively parses the rules in order to determine if the goal is reachable. The model is not recent, but is still widely used, for example in popular software frameworks like JESS (*JESS homepage*).

However, this approach is subject to the criticisms that symbolic reasoning has received in the past years. These include the philosophical argument of John Searle (1980), who has shown with his "Chinese Room" experiment that the coherent manipulation of symbols is not equal to a real understanding of what these symbols mean. A more heuristic criticism is Rodney Brooks's (1991): he acknowledges the failure of the symbolic approach as a fact, and proposes a more reactive and incremental strategy with its Subsumption Architecture. The architecture is not centrally controlled but includes a number of quasi-independent software layers (e.g. Standup, Simple walk, Force balancing, etc.) which are hierarchically organized and communicate asynchronously. The higher-level layers are built on the lower layers or can inhibit them. The rich behaviour of the system results from the simple interactions of the layers.

On a different level of abstraction, agent-oriented software design is becoming extremely popular. We refer to Wooldridge and Jennings' definition of agency: an agent is "... a hardware or (more usually) software-based computer system that enjoys the following properties:

- autonomy: agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state;
- social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language;
- reactivity: agents perceive their environment, (which may be the physical world, a user via a graphical user interface, a collection of other agents, the INTERNET, or perhaps all of these combined), and respond in a timely fashion to changes that occur in it;
- pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behaviour by taking the initiative." (Wooldridge & Jennings 1995, p. 2)

The belief-desire-intention (BDI) architecture (Georgeff et al. 1999; Wooldridge & Jennings 1995), which is discussed in Chapter 4, is gaining interest as a framework to build intelligent autonomous agents. Several attempts have been made in order to make BDI architectures better suited to the generation of social behaviours (Dignum et al. 2000; Panzarasa, Norman, & Jennings 1999; Kalenka 2001; Guye-Vuillème & Thalmann 2001).

1.3.2 Artificial Life and Action selection

In contrast with symbolic reasoning, Artificial Life proposes a number of bottom-up approaches, frequently based on evolution and learning techniques, in order to simulate living systems. A good overview of the achievements and open problems in the field is given by Bedau et al. (2000). Many of these contributions deal with populations of simple organisms like insects and are not directly applicable to small groups of humans. Nevertheless, research on such topics as the evolution of social behaviour through social learning (Noble & Franks 2002), which consists for example in copying a randomly-chosen neighbour, are potentially interesting for the simulation of human social behaviour. In a related context, Channon and Damper (1998) have used a connectionist approach and argue that evolutionary emergence is the best technique to obtain socially intelligent agents. They describe a multi-agent system using neural networks and genetic algorithms that generates social behaviours such as cooperation and competition. However, the type of behaviour produced with this approach still appears extremely basic.

Research in the subfield dealing explicitly with the action selection problem has also produced several interesting models. The early conception of Lorenz (1950) for simulating the growing motivation that brings an organism to action is particularly inspiring. His psycho-hydraulic model is based on a analogy between the processes prioritising the organism's needs and drives and the circulation of a fluid. More recently, Tyrrell's work (1993) on free-flow hierarchies has produced good results in the simulation of the adaptive behaviour of animals immersed in dynamic environments. In free-flow hierarchies, a number of behavioural nodes are organised vertically so that activation or inhibition energy is spread from the top-level perceptual input to the lower motor control nodes via weighted connections. Contrary to the classic hierarchical planner, there is not a threshold automaton at each level generating a winner-take-all operation, but the spread of activation is unrestricted. De Sevin, Kallmann and

Thalmann (2001) are currently exploring Tyrrell's model for the simulation of human behaviour.

In a hybrid approach, Maes (1989) proposes a non-hierarchical action selection mechanism combining characteristics of traditional planners and reactive systems. An agent is here defined as a network of competence modules, which exchange activation and inhibition energy. On one side of the network is entered the energy corresponding to the agent's goals, on the other side, the energy corresponding to the state of the environment. After some time, the module where the most energy has accumulated is selected. Depending on the requirements, the agent's behaviour emerging from the interaction of the modules can therefore be easily made more reactive or more goal-oriented. In spite of the vivid debate between Tyrrell and Maes, we consider both models valuable for the creation of realistic virtual humans.

1.3.3 Social Simulation and Social Agents

A good overview of the simulation techniques used by social scientists is given by Gilbert and Troitzsch (1999). Several different paths have been explored, but two main approaches emerge from recent research: cellular automata and social networks simulation.

A cellular automaton consists in a number of identical cells arranged in a regular grid. In social simulation, these cells typically represent the members of a population. Each cell can be in a number of specific states, like dead or alive, but these states can also represent attitudes, beliefs, etc. Using a set of basic rules describing how a cell is influenced by its immediate neighbours, the state of every cell can be modified every time step. With this approach, such phenomena as the evolution of culture, the dissemination of social norms and ideas in a population, or the dynamics of conflict, can be simulated in a powerful way. For instance, Lustick (2000) used a cellular automaton to explore the formation of collective identities.

Social networks simulation is related to cellular automata but relies on the mathematical theory of graphs, and is better suited to the study of group behaviour. In a social network modelled as a graph, the nodes represent individuals and the edges represent social relationships. Like in a cellular automaton, the state of a node can be modified by its neighbours. However, the influence of the nodes is not only local but is related to the characteristics of the graph. Simulated social networks have various structural forms and a number of specific properties: size (the number of nodes in the network), connectivity (the number of connections per node), neighbourhood (the template for the neighbourhood of each node), rule scheme (the functions that describe the interaction between nodes), and updating methods (used to change the state of the nodes). Stocker, Green and Newth (2001) have used this approach to simulate the development of consensus and cohesion in societies. A well known model of group stability, based on a fully connected social network, has been proposed by Carley (1991). Her model relies on the idea that interaction leads to shared knowledge and that relative shared knowledge leads to interaction.

Interestingly, a research community dedicated to the conceptualisation and building of synthetic social agents is emerging from the social simulation field. Carley and Newell (1994) at Carnegie Mellon University, have been among the first to construct a consistent view of the social agent based on the agent's information processing capabilities and the type of knowledge handled. According to Carley and Newell, social agents, which are immersed in complex environments, need to gather important information from other agents. For this purpose, they are driven to initiate and maintain social relationships, and may be mutually supportive in the execution of actions. In addition, social agents must handle complex and differentiated knowledge about the self, task domain and environment, which are specific to a given cultural and historical context. More recently, Kerstin Dautenhahn (1998) has been very prolific on the topic of the design of socially intelligent agents and has provided interesting guidelines. Her work is focused on creating agents that are able to recognize each other, establish and maintain relationships as well as collaborate, but much attention is also given to other important aspects such as empathy, embodiment and historical grounding (Dautenhahn 1999).

A number of associated contributions are relevant to our work, for instance Conte and Castelfranchi's (1999) proposals on the problems of social norms representation, cognitive emergence and agent rationality. The development of benevolent agents that can adopt other agents' goals spontaneously is an example of an alternative type of rationality which could prove useful in the simulation of social interaction. On a similar topic, Kalenka (2001) proposes a framework enabling the modelling of social interaction attitudes ranging from purely self-interested to purely altruistic. In addition, Boman and Verhagen's (1998) work on social norms acceptance and learning, Sichman's (1995) model of inter-agent relationships based on dependence relations, and Senger's (1997) exploration of action expression in social agents, are particularly interesting contributions. In the domain of robotics, Mataric's (1997) work on the generation of adaptive group behaviour and social behaviour learning is also relevant.

On a more technical level, we would like to mention SDML, which is an XML-based language for the rule-based modelling of agent interactions in organizations provided by Moss et al. (1998), and SWARM, a toolkit for the building of multi-agent simulations which is commonly used by social scientists (*SWARM homepage*).

1.3.4 Behavioural Animation and Embodied Agents

The concept of Behavioural Animation is introduced by Reynolds (1987) in order to describe a type of procedural computer animation in which autonomous characters determine their own actions and are able to improvise according to the situation. From a computer animation point of view, the main advantage is that designers do not need to specify all the details of every motion, however it requires an adequate modelling of the characters' perceptive and motivational apparatus. The system design issues in Behavioural Animation are therefore fundamentally linked to research in autonomous agents. Reynolds' (1987) classic "boid" model illustrates this approach for the animation of birds and fishes. In order to generate flocking behaviours, a number of autonomous creatures called "boids" alternatively use three simple steering actions according to the position and velocity of their nearby flockmates.

Behavioural Animation has been used to generate animations involving humans, like in research by Bécheiraz and Thalmann (1998) or Monzani (2002). Although these contributions are clearly related to our work, they emphasize different aspects of human behaviour like the simulation of emotional reactions and virtual storytelling. Bécheiraz and Thalmann's work includes some interesting proposals for the simulation of social interaction, like a list of standing postures for face-to-face encounters, but the topic is not systematically explored, e.g. the proposed postures are limited to male-female interactions.

The problem of simulating human groups and crowds in virtual environments has also received attention in the behavioural animation community (Bouvier & Guilloteau 1996; Musse & Thalmann 1998; Ulicny & Thalmann 2001; Villamil, Musse & de Oliveira 2003). Particle systems are commonly used, like in the work by Bouvier and Guilloteau (1996), as well as approaches involving scripted animation (Musse & Thalmann 1998). Interestingly, the research focus is shifting from a paradigm where the group or the crowd is centrally controlled by an abstract entity or by an all-mighty leader, to a model where collective behaviour results from the actions of the individuals. We share the latter view, which is consistent with our definition of social systems. In this context, Ulicny & Thalmann (2001) propose a layered architecture where a combination of behavioural rules and finite state machines allows the animation of emergent crowds.

In addition to these recent contributions oriented to the production of 3D content, we would like to reference the work from crowd researcher Clark McPhail. Using his GATHERINGS program (McPhail, Powers & Tucker 1992), he generated extremely interesting animations showing how individuals control their spatial relationship with others in the context of moving groups and temporary gatherings. The produced visuals may not be as spectacular as more recent works, but they are strongly grounded on social sciences.

In the framework of embodied agents, a number of systems incorporate a social dimension in the behaviour of their agents. The OZ project (Bates, Loyall & Reilly 1992) defines an agent architecture which supports goal directed and reactive behaviour as well as emotions and social behaviour. The importance of personality and emotions for the generation of believability is here emphasized. In a related approach, Rousseau and Hayes-Roth (1998) propose a social-psychological model for synthetic actors, based on personal traits, moods and attitudes. A number of rules are defined that specify which behaviour must be triggered according to the personality and current state of the agent. Rizzo et al. (1999) use a planning architecture to provide believable social agents exhibiting different goal-based personalities, like help-givers. The work by Pelachaud and Poggi (2002) on multimodal embodied agents is particularly interesting in its nonverbal communication aspects, for example the use of multimodal lexicons or the synchronization of the different modalities. Among other contributions, Prendinger and Ishizuka have underlined the importance of social role awareness in animated agents (2001), and propose a model of social relationship evolution (2002) based on the agents' affective interaction histories.

Before concluding this section, we would like to mention a very popular video game called "The Sims™" from designer Will Wright (*The Sims homepage*). The Sims™ is a so-called "people simulator": users can create 3D human characters having a specific physical appearance, a living environment, a personality, social relationships, etc. The characters are semi-directed: they can be controlled by the player who can trigger actions or make two characters communicate, but when left to their own devices, they automatically take action in order to satisfy their fundamental needs, like the need for food or hygiene, but also the need for social interaction. Like in our work, the emphasis is put on nonverbal activity: when two characters interact, they position themselves face-to-face and a number of gestures and facial expressions are triggered, but verbal expression is only symbolized. Although The Sims™ and our work have many objectives in common, we don't share the view that social interaction can be solely defined in terms of needs, and we try to take into account the existing body of knowledge rather than making *ad hoc* choices. Typically, the "proxemic" aspects of nonverbal social interaction, i.e. the way people use space to communicate, are ignored by the game's designers; contrary to what can be observed in real life, the characters are, for instance, always positioned straight in front of each other when interacting.

1.4 General Approach

It appears in the literature we have selected, that our work comprises a tension between a "bottom-up" and a "top-down" approach. Bottom-up approaches, like Artificial Life, aim at creating the conditions for rich behaviours to emerge from the interaction of basic components in a complex system. The technique is very appealing since it is at the same time simple and powerful. However, its main drawback is that the behavioural outcome cannot be finely controlled. For example, when a precise output is needed, which is often the case when the objective is to simulate behaviours observed in real life, it can be extremely difficult to know what changes should be introduced in the system in order to produce it. Frequently, long sessions of trials and errors are required in order to determine what should be modified or to train the system to produce the adequate outcome.

On the other side, the top-down approach, where high-level behavioural specifications are given to the system, produces only desirable and predictable results in normal conditions. Unfortunately, the writing of the behaviour's specification is often tedious and the produced outcome can sometimes be perceived as lacking adaptability in the same way as scripted animations would. The choice of basing our work on social sciences research favours a top-down approach, since a major part of the available data consists in observations which describe the phenomenon itself, or the correlation between the phenomenon and some statistical variables, rather than the precise mechanisms producing the phenomenon.

Consequently, we adopt a mainly top-down approach which includes some hybrid aspects. It consists in providing to a simulation system some high-level specifications describing the behaviour to produce or the characteristics of the agents in charge of producing it, and in designing the system so that conflicting or reinforcing interactions between some parts of the specification alter the regularities in the system output. Ideally, the degree to which the initial specification is altered should be controllable, e.g. by increasing or decreasing the number or the impact of the interactions between the internal components. In addition, the way it is altered should be meaningful for an external observer. When modelling a single agent, the interacting components can be, for example, the differentiated aspects of its personality; when simulating group behaviour, group members interact and influence each other to produce a behaviour which is more than the addition of every dyadic relationship.

The issue of emergence has been widely discussed, but little agreement exists as to what exactly are emergent systems and how they can be identified. According to Ronald, Sipper and Capcarrère (1999), the degree of surprise of an expert observer confronted to the behaviour of the system is a crucial criterion. More precisely, it can be said of a system that it has emergent properties when a "cognitive dissonance between the observer's mental image of the system's design" and his/her "contemporaneous observation of the system's behaviour" occurs. In our approach of group dynamics, the degree to which a simulation system is emergent can be calculated as the difference between the specification provided in terms of interaction rates and the observed output in terms of interaction occurrences.

1.5 Organization

This document is organized as follows. In Chapter 2, we undertake the identification of the required functionalities for the visual simulation of nonverbal social interaction in virtual environments. For this purpose, we present our observations of how real people choose to interact nonverbally in virtual environments, based on a small-scale experiment we have carried out at EPFL.

Chapter 3 discusses the project's requirements in terms of computer graphics and animation, and presents a number of technological components allowing the visualization of rich nonverbal behaviours. The emphasis is put on standardized face and body animation, on the synchronization of the verbal and nonverbal modalities, and on the integration of these components with agent technology.

The issue of modelling social agents is tackled in Chapter 4. In this section, we review some of the underlying assumptions behind the mainstream model of agency, present a number of sociological concepts which can constitute an alternative approach, and, on the basis of these concepts, propose a high-level architecture for the modelling of individual social agents.

In Chapter 5, we present our work on the simulation of small groups dynamics. In addition to a XML language for the characterization of group members and a tool for the control of group interactions, we present two main contributions: a model of action selection for group members and a model of interpersonal relationship development. It is explained how the interrelation of these two models results in the construction of realistic sequences of social interactions.

Chapter 6 describes how these sequences of social interactions can be visually simulated in virtual environments. We present our work based on well known models in the field of Nonverbal Communication, which comprises, among other contributions, a number of dedicated parametric animations, a format to specify the meaning of nonverbal actions and a mechanism triggering nonverbal actions according to the specific interactional context.

Finally, in the conclusion, we discuss the limitations of our work and suggest some directions for future research.

Chapter 2 • Initial Experiment: People Interacting in Virtual Environments

2.1 Introduction

Social interaction is an extraordinarily rich process which involves almost all human capabilities. In real-life, when two persons meet in a face-to-face situation, a very high number of communication channels are simultaneously activated and numerous messages - possibly based on very different codes - are exchanged in a short time (Argyle 1988).

It is doubtful that a simulation system will ever be able to reach such a degree of complexity and completeness. In any case, it is clear that, in the current state of the field, not all dimensions of such a "total social fact" can be simultaneously handled. Thus, researchers willing to simulate social interaction have to make a number of important decisions about which aspects to focus on or not, which mechanisms to recreate, mimic or temporarily let aside.

In our research, the most important criterion we have used in order to make these decisions is, logically, the available body of knowledge. As previously mentioned, we have in priority tried to recreate real-life codes and mechanisms identified by social scientists as particularly important for the conveying of meaning in a social context. Yet, it appears that a complementary source of information can prove useful: the observation of how real persons choose to interact in a constrained environment which is similar to the simulation environment.

This issue is important because it helps understand and take into account the impact of the media limitations on the simulation. We postulate that people, whether in real or virtual environments, are driven to produce specific social meaning and adapt the way they express it (e.g. their social identity, the nature of a relationship, etc.) according to the media. Obviously, the limitations of the tools provided for interacting have an important impact on the participant's behaviour, but, beyond this aspect, important knowledge can still be gathered: the favoured communication channels in such environments, the type of content for which these channels are used, the limitations that appear the most restricting to the participants, the strategies they invent to try to by-pass these limitations, etc.

Since few evaluations of people interacting in virtual environments using embodied characters were available, we implemented the necessary tools and conducted our own experiment in the hope that it would point to the features that are most important to replicate. In a more general way, we thought it would serve as a general source of inspiration for the rest of the research. In addition, the implemented tools are valuable in the context of Collaborative Virtual Environments (Guye-Vuillème et al. 1999).

For this project, we exploited LIG[1]'s IRIX-based flexible framework for the integration of virtual humans in the Shared Virtual Environments, called VLNET (Virtual Life Network) (Capin et al. 1997).

2.2 Development of a Nonverbal Communication Application

2.2.1 Introduction

Virtual environments are often referred to as cold and dehumanised places by the users. Static representations are also generally considered as lacking emotions (Vilhjálmsson & Cassell 1998). The development of a nonverbal communication application is required for our experiment but may also have the advantage of greatly improving the quality of the experience lived by the participants.

In this section, we first discuss the basic facilities that must be provided in order to allow reasonably rich interactions to take place in Shared Virtual Environments. The main functions fulfilled by the different elements of the simulation are identified and analysed; the first element to be considered is the user embodiment.

2.2.1.1 Embodiment

The embodiment fulfils several important functions in virtual environments:

- the visual representation of the user,
- means of interaction with the world,
- means of sensing various attributes of the world

It becomes even more important in multi-user environments, as participants' representation is used for communication. Embodiments are required for:

- perception (to see if anyone is around),
- localisation (to see where the other person is),
- identification (to recognise the person),
- visualization of others' interest focus (to see where the person's attention is directed),
- visualization of other's actions (to see what the other person is doing and what is meant through gestures),
- social representation of self through decoration of the avatar (to know what the other participants' identity or status is)

Yet, the type (realistic to metaphoric) and complexity (based on simple geometry to fully deformable) of the embodiment are widely discussed issues in the field. Most of the existing

[1] Former name of VRLab

systems still use simple embodiments for the representation of participants in the environments. We consider that more complex embodiment is necessary for bodily communication because the users must be able to identify a basic set of (animated) limbs on others' avatar, and for the improvement of natural behaviour within the environment. The users' more natural perception of each other (and of autonomous actors) increases their sense of being together, and thus the overall sense of shared presence in the environment. Moreover, the use of an articulated structure corresponding to a skeleton is an efficient technique and has become common in computer animation. These elements have led us to use a complex human-like representation. Low-level facilities for handling such embodiments are provided in VLNET (Capin 1998).

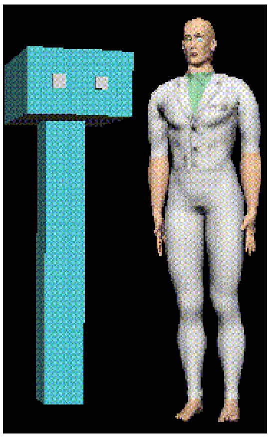


Figure 1 Two examples of embodiment: a "blocky" Mister T and one of our complex models Peter

2.2.1.2 Navigation

Free navigation within the virtual environments, i.e. the user is not constrained to a set of predefined locations, is another important functionality to provide in the context of embodied social interaction, since it allows users to:

- approach or avoid each other,
- communicate their interest focus (through body orientation),
- signal their desire to interact (by moving in front of another avatar),
- interrupt an interaction (by abruptly leaving),
- communicate about their identity and relationships (through choice of distance and angle for the interactions)

The simple mouse-based navigation system integrated into VLNET's core fulfils these requirements.

2.2.1.3 Verbal Communication

Although verbal communication is an important aspect of social interaction, it is not the primary interest in our research. Thus, listing the numerous types of information that can be exchanged using the verbal channel would be beyond the scope of this document. Verbal communication has been integrated in the project mainly to preserve the seamless and natural character of subjects' interactions. Numerous desktop solutions are readily applicable to communicate verbal content.

2.2.1.4 Face and Body Expressiveness

Face and body expressiveness is important to include in shared virtual environments because it is a fundamental component of face-to-face interaction. Corraze (1980) proposes to distinguish between three types of convoyed information:

- information about the affective state of the sender,
- information about his/her identity,
- information about the external world

Facial expressions (e.g. smiling) and body postures (e.g. relaxed) are typically used to communicate the affective state of the sender but several gestures (e.g. insult) can also be used to convey strong emotional messages. Information about the user's identity is generally passed through a number of gestures which are specific to a group or subculture. Hands and arms movements serve to give information about the external world (e.g. pointing).

In addition, it has been shown that facial and bodily activity plays a central role in the regulation of speech. It can be used by people as a means to signal importance or that the speech is finished, for example. Several studies give good examples of the high level of intricacy between the speaker's speech and actions, and the listener's nonverbal behaviours (Weitz 1974).

Since this particular dimension of face-to-face interaction was unavailable to VLNET users, we developed a suitable nonverbal communication application. The following sections describe our choice of implemented face and body actions, the user interface and the implementation in VLNET.

2.2.2 Selected Actions

For the experiment, we wanted a relatively small number of gestures and postures, so we decided to try to identify a basic "palette" of actions, which is a difficult task because nonverbal communication does not work as a linguistic system. The following criteria were used to select the actions described in Table 1:

- documented in scientific papers,
- basic action, commonly used, expresses simple idea,
- different enough to compose a "palette" of actions,
- can be understood in many places/cultures,
- can be performed in the standing position,
- a graphical representation of the action was available

The body postures and gestures come from a classic and commonly used sample of nonverbal actions, first developed by Rosenberg and Langer (1965). The postures we have selected illustrate very well the classic four fundamental postural attitudes described by W. James in which the positions of head and trunk are essential: attitude of approach with the body bent forward ("Attentive"), attitude of rejection with the body turned away ("Rejection"), attitude of pride with the expansion of head, trunk and shoulders ("Determined"), attitude of prostration with the head bent and the shoulders falling ("Insecure") (Corraze 1980). The hand gestures were chosen because their cultural and geographical distribution has been intensively studied, e.g. by Morris et al. (1979). Finally, the sources of facial expressions are Miller's (1976) and Ekman and Friesen's (1967) work.

Postures / Expressions		**Gestures / Mimics**		
Face	**Body**	**Head/Face**	**Body**	**Hand / Arm**
Neutral	Neutral	Yes	Incomprehension	Salute
Happy	Attentive	No	Rejection	Mockery
Caring	Determined	Nod	Welcoming	Alert
Unhappy	Relaxed	Wink	Anger	Insult
Sad	Insecure	Smile	Joy	Good
Angry	Puzzled		Bow	Bad

Table 1 Selected actions, classified by posture/gesture and part of the body

New actions can be easily added by users without programming knowledge and without recompilation of the application.

2.2.3 User Interface

In order for our solution to be usable with a desktop configuration, we decided to develop a dynamically generated 2D interface allowing the user to select among predefined actions specified in the application's configuration file. This approach is less appropriate for actions that are not always under conscious control, e.g. the postures, but it seemed to us the best compromise between practical constraints and our desire to include this aspect of human communication in a desktop environment.

In order to fulfil the need for an intuitive and easy to learn user interface, it was decided to use image buttons displaying a snapshot of the actual move and a textual label describing the idea or state of mind expressed by the action.

We decided to work with three windows: the posture, gesture and control panels. The panels offer a global view of all actions available, with clickable image buttons. They are constituted of several sections containing the actions classified by part of the body and "emotional impact" (positive, negative, neutral). The user can also parameter the speed of execution of the action, and use keyboard shortcuts to run them. The high degree of organisation of the actions combined with the fact that all actions are immediately activable, allow the user to rapidly find and execute the action that best fits the situation.

In "compact mode", the panels can be automatically attached and scaled with the VLNET view window for convenience. A "mood setting" (cool, normal, stressed) modifying the speed and frequency of gestures, and the possibility to automatically follow other participants, have also been added.

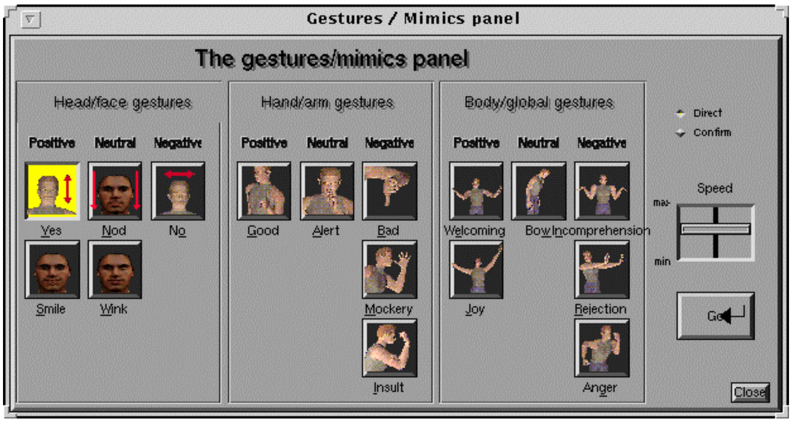


Figure 2 The gesture panel

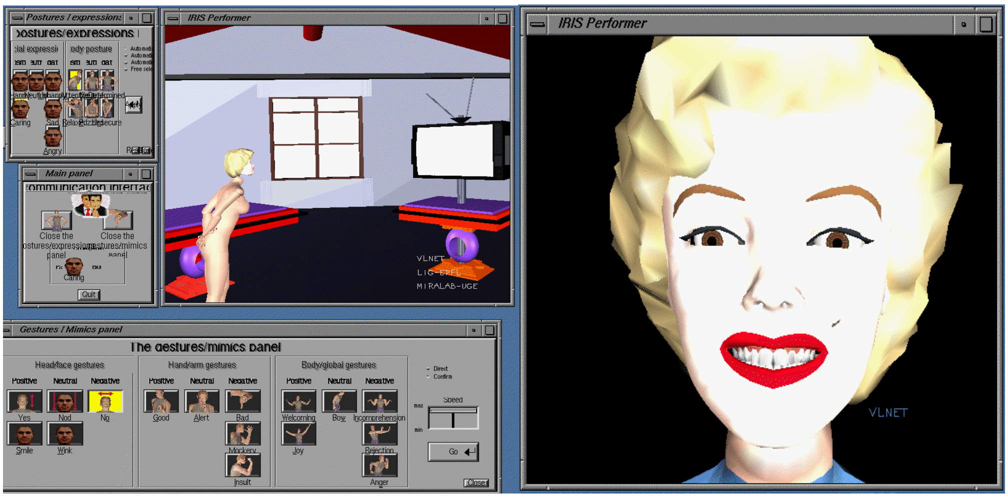


Figure 3 Example of use of the nonverbal communication application

2.2.3.1 Implementation Overview

The implementation is highly related to the VLNET software architecture which is described in more details by Capin (1998).

In our solution, the main VLNET process executes the low-level simulation tasks and provides services for the basic elements of virtual environments to the external programs, called drivers (Capin et al. 1997). The VLNET core consists of logical units, called engines. The role of the engine is to encapsulate one main function of the virtual environment in an independent module, and provide an orderly and controlled allocation of virtual environment elements. Moreover, the engine manages this resource among various programs which are competing for the same object.

Drivers provide the simple and flexible means to access and control all the complex functionalities of VLNET. Each engine provides a shared memory interface to which a driver can connect. The drivers are spawned by the VLNET Main Process on the beginning of the session. From the VLNET system point of view, the nonverbal communication application is a Facial Expression Driver, using the MPA (Minimal Perceptible Actions) format which provides a complete set of basic facial actions allowing the definition of any facial expression, and a Body Animation Driver which controls the motion of the user's body.

For the control of the virtual human body posture animation, an articulated structure corresponding to the human skeleton is used. Structures representing the body shape are attached to the skeleton, and clothes may be wrapped around the body shape. We use the HUMANOID articulated human body model with 75 degrees of freedom without the hands, with additional 30 degrees of freedom for each hand (Boulic et al. 1995). The skeleton is represented by a 3D articulated hierarchy of joints, each with realistic maximum and minimum limits. The skeleton is encapsulated with geometrical, topological, and inertial characteristics of different body limbs.

Attached to the skeleton, is a second layer that consists of blobs (metaballs) to represent muscle and skin. During runtime the skin contour is attached to the skeleton, and at each step is interpolated around the link depending on the joint angles. From this interpolated skin contour the deformation component creates the new body triangle mesh. Thus, the body information in one frame can be represented as the rotational joint angle values.

The system, VLNET core and nonverbal communication application driver with complex embodiments, is designed for SGI workstations: low-end models (e.g. O2) are sufficient for three or less participants, but more powerful workstations (e.g. OCTANE, ONYX) are necessary for a higher number of users. Detailed performance data has been documented by Capin (1998).

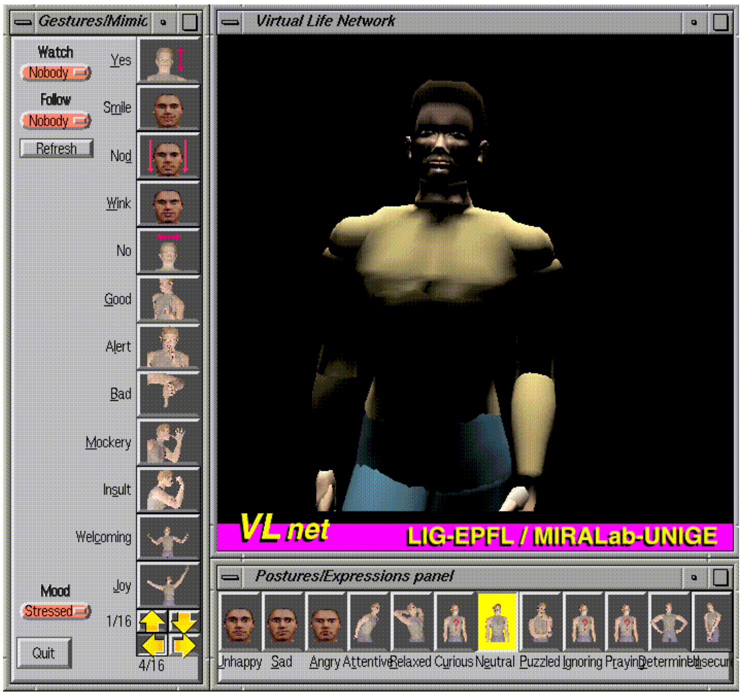


Figure 4 The user interface used in "compact mode"

2.3 Experiment

The first decades of research in the Nonverbal Communication field have seen a wide use of laboratory experiments. Nowadays there is an increasing preference among psychologists for observing real and spontaneous behaviour (Argyle 1988). In the Collaborative Virtual Environments field, the use of ethnographic methodology has given good results for evaluating applications and identifying typical practices (Bowers et al. 1996). These considerations motivated our choice of evaluation method.

2.3.1 Experimental Setting

In order to encourage spontaneous behaviours and to reduce the impact of the researcher on the results, participants were let free to act and interact as they chose. Six participants took part in the study, and two hours of interaction were recorded and analysed. Our analyses were qualitative in nature; careful observations of their interactions were taken, and their impressions were gleaned from a survey conducted at the end of the study. Because we wanted to have results that could guide us in the next developments, we didn't try to “prove” an hypothesis but to identify crucial issues and behaviours.

One of our main interests in carrying out this study was to establish if the users, using the nonverbal tools at their disposal, could replicate their relationship with other participants. Thus, the degree of intimacy with each other has been our main criterion in selecting the subjects for the experiment. We chose six participants, none of whom were computer scientists:

- two females: R and L,
- four males: J, J2, T and R2

Grouped in pairs, two were very familiar with each other (R and J2), two were acquaintances (J and L) and two were strangers to each other (R2 and T).

After an introduction to the system, they were given total freedom of action, being allowed to talk with each other or stay silent, explore the scene or stay at the same place, use nonverbal communication or not. Three systems were at their disposal for interacting: - a navigation system allowing their avatar to walk freely in the environment, rotate, etc. - the nonverbal communication application with its thirty actions - a microphone and headphones for verbal communication. The only constraint was that the nonverbal communication application was voluntarily deactivated by the experimenters on a regular basis. Every participant had been assigned a different human-like embodiment corresponding to its gender identity. The scene we used represented a public square with a bar in its centre and was chosen for its public and socially oriented characteristics.

SGI OCTANE/SI (175 MHz) and OCTANE/MXI (195 MHz) workstations on a 100baseT network were used for the experiment.

2.3.2 Evaluation

Here is a summary of the observations and comments gathered during and after the experiment. It includes ideas either relevant to the simulation of social interaction or useful for the evaluation of the solution itself.

2.3.2.1 Use of the Nonverbal Interface

The users generally had no difficulty in using the interface. After a couple of minutes, all were used to it and J2 and R even started to "play" with it, trying to run several actions simultaneously. The participants used some actions a lot more than others, the main point being that they used many more gestures than postures. A posture was very often chosen at the beginning of the interaction, but it stayed a long time as the participants didn't think of changing it. This can be explained by the fact that postures are often chosen unconsciously.

For a similar reason, the speed setting was not used very often and was rated "*not very convenient*" (J) in the survey.

The study was divided into sessions during which the nonverbal communication application was active or inactive. In the survey, the periods without nonverbal communication were rated as "*boring*" by the majority of users. Typical expressions used by the participants to describe the influence of the application on their experience are: "*it was funny*" (R, L and J), "*added something*" (T), "*the whole scene seemed more life-like*"(J). The inclusion of emotional content was rated by all users as "*useful*" or "*pleasant*".

Because we chose actions that don't need to be highly synchronised with speech, the users had few problems of this kind. It is mainly the "yes" and "no" gestures that they wanted to run at the same time as the corresponding words. If the delay of the nonverbal signal was long at the beginning of the experiment (several seconds), it significantly reduced when they got used to the interface (approximately 1 second). Interestingly, the "attentive" posture was frequently used as a way to indicate to the speaker that one was listening to him, and the "puzzled" posture when questions were asked. This confirms that regulating speech is also a very important function of nonverbal communication.

Another fundamental need emerged from the participants' impressions collection: the presence of bodily feed-back. Without being able to "feel" the posture of their avatar, J, L, R and J2 strongly asked for the possibility to view their own body during the experiment. However, this solution could take away some of the immersion feeling because the user can consider himself/herself as totally exterior to the situation. A strategy they used was to ask other participants about their own appearance. The simulation of proprioception is a difficult challenge for virtual reality researchers, but crucial for a quality immersion in the virtual environment and control of the avatar.

The caricatured aspect of many gestures and postures was also emphasized in the survey. Through motion capture and the use of actors, more realistic actions could be produced. Nevertheless, the probability is high that any predefined action would be considered caricatured, or would not be understood easily enough if the visual clues were to be weakened. The main point is that predefined actions cannot, by definition, be finely adjusted to the specific ongoing interaction. But according to the users report, this caricatured aspect of actions was disconcerting only at the beginning of the experiment. Then, the users got used to it, and used these actions for their symbolic meaning.

2.3.2.2 Reproduction of the Real World Social Relationships

It is interesting to notice that the users have been able to reproduce, through the mechanisms of nonverbal communication, their relationship of the real world in the virtual environment.

We observed that the subjects that didn't know each other before the experiment (R and T) situated themselves at a bigger interactional distance than the ones who were familiar, and this is typical of what the study of Proxemics has showed. Moreover, they carefully avoided all aggressive gestures when the others (who knew each other) used several times the "mockery" gesture or the forearm jerk.

At another level, the nonverbal communication application allowed them to also respect the formal structure of social interactions. At the beginning of the interaction, they all used one of the actions to greet the other one ("Bow", "Welcoming") and signal that they were ready to begin the exchange. The end of the interaction followed the same logic and was always confirmed by nonverbal means. The normative sanction produced when someone doesn't respect these rules in real life showed up.

For instance, R was speaking with J2. R suddenly decided to explore the world and abruptly left J2. J2 became angry and used verbal and nonverbal means (anger and insult gestures) to express it. R came back and they left to explore the world together.

Many other elements confirm the reproduction of the real world social relationships. During the experiment, the avatars of J (male) and L (female) collided with each other. They naturally apologised and then laughed of the experience. Later, the avatar of J and L were very close, nearly touching each other in a position that could have been interpreted as very intimate. A strong emotion was noticed on the participants, first in the form of uneasiness and then laugh. This behaviour is typical of the relationship between J and L: they have different gender identities and don't know each other very well. The movements and positions of their avatar weren't "free", as the consequences would have been similar if the scene had occured in real life.

A last example illustrates this "real" effect of "virtual" interactions: during the experiment, J2 became really angry because R wanted him to do something that he didn't want to. R refused to speak for a moment but used the "forearm jerk" gesture in a totally sincere way.

2.3.2.3 Missing Features

In the survey, all users had difficulties in identifying what useful gesture or posture was missing. They admitted that their method was to examine what was at their disposal and use it, rather than searching what would be best suited and check if the application had it. What has been strongly requested is the ability to touch the other avatars, tap, punch or simply shake hands. This underlines the importance of actions involving physical contact.

However, the main limitations did not come from the composition of the "palette" of actions itself. According to the subjects' comments, what weakened the realistic aspect of their interactions was a "*strange feeling*" (J) of discontinuity. The sentence *"It's not funny, you're not moving !"* (R) is typical of this record: in real life, you cannot stop communicating. While interacting in the virtual environment, the subjects always wanted to decode signals that were not present or just suggested. During the debriefing, they identified several mechanisms whose integration would have suppressed or weakened this effect:

- Lips movements should follow speech.
- Hands movements should illustrate speech.

- Head's and eyes' orientation should be properly controlled.
- Facial expression should appear less static.

The automatic handling of such actions would necessitate the use of advanced tracking facilities of users' face and body, and/or the implementation of intelligent modules controlling parts of the model according to a set of predefined or learned preferences.

At the conclusion of the experiment, it appears that we have given the users the possibility to send important messages to their interlocutors that they couldn't send before, but in a rather raw and limited way.

Figure 5 J and L interacting at the bar

2.4 Conclusion

The carrying out of this initial experiment allowed the collection of a number of learned lessons valuable for the continuation of our core research. First, it appears that the evaluation's results are generally consistent with Nonverbal Communication literature. A number of classic nonverbal channels have been extensively used by the participants in order to convey, through the virtual environment, information about themselves or about their relationships:

- interpersonal distance,
- interaction angle,
- facial expressions,
- gestures,
- postural shifts

It is confirmed that these dimensions are as crucial in virtual environments as in real life, and need to be addressed by a system aimed at the simulation of social interaction.

In addition, the evaluation has underlined the importance of some specific aspects of nonverbal social interaction, which led us to identify the following requirements:

1. **Speaker and Listener roles must be reinforced**. It is important to associate specific nonverbal behaviours to each interlocutor so that a clear distinction can be made between them. Not only synchronized lips movement must accompany speech, but also a number of hand gestures called “illustrators”. The Listener also has to exhibit behaviours of its own, such as the use of an adequate posture and gaze, but also a feed-back channel constituted of such gestures as head nodding.

2. **Interaction phases must be taken into account**. In the virtual environment, participants have used different codes to initiate, maintain or terminate an interaction. This feature must be simulated by using different catalogues of nonverbal actions for each phase.

3. **Gestures involving interpersonal physical contact are important to trigger under adequate circumstances**.

4. **Continuous nonverbal activity must be ensured**. Whatever the situation, a minimal amount of nonverbal activity must be observed on each embodiment, under the form of specific facial expressions, eyes blinking, such gestures as scratching its head, automatic wrist movements, etc. These kinds of actions are known as “adaptors”.

Finally, it is interesting to note that the experiment has given us a foretaste of what our final simulation would be like, which is an indirect but certainly valuable contribution. Moreover, it is always very inspiring to observe how real people are able to adapt to new situations, and invent strategies allowing them to achieve rich social interactions in constrained environments.

Chapter 3 ● Technology for the Visualization of Nonverbal Behaviours

3.1 Introduction

As discussed in the preceding chapter, nonverbal behaviours play a central part in human face-to-face interaction. In order to properly simulate this type of interaction in virtual environments, adequate technology must be used to visualise the facial and bodily activity associated to it. In particular, the following functions must be addressed:

- rendering of the interaction's context: a 3D scene,
- rendering of the embodiments: 3D face and body models,
- facial animation: face model deformation,
- body animation: limbs movements through body model's joints angles manipulation,
- body translation: change of body model's location in the scene,
- efficient streaming of animations, including data compression

The previously discussed VLNET system is an example of suitable solution. Yet, it is based on proprietary technology. In order to maintain the generality of our contribution, we will, in this chapter, focus on technology which allows the real-time visualization of embodied interactions and which is based on an established standard: MPEG-4.

Using standardized technology has several advantages, among others: the developed components can be easily shared and reused, they are based on verified techniques and algorithms, the implementations can be properly compared and validated.

MPEG-4 is an ISO/IEC standard developed by MPEG (Moving Picture Experts Group), which provides the standardized technological elements enabling the efficient integration of the production, distribution and access to multimedia content and interactive graphics applications. One sub-part of it, the MPEG4-SNHC (Synthetic-Natural Hybrid Coding) standardization, includes the specification of face and body animation (version 1 of the specification for face animation, version 2 for body animation). The previously described low-level requirements are addressed (among others) in full MPEG-4 compliant systems.

However, MPEG-4 does not provide any high-level functionality for the definition of advanced multimodal behaviours allowing the simulation of rich social interactions. We have identified a number of features that are required:

- organization of multiple face and body animations in time,
- motion blending for simultaneous actions,

- synchronisation with speech,
- parameterisation of predefined animations,
- integration with real-time algorithmic animations, such as pointing or walking

In this chapter, we first introduce face and body animation in MPEG-4 as well as the compliant tools we have implemented. Then, we present AML - the Avatar Markup Language, an XML-based language that we have developed, with the University of Geneva (Miralab) and the Imperial College, in order to fulfill the high-level requirements just described. Finally, we discuss the integration of our MPEG-4 face and body animation components and of AML in a target MPEG-4 platform: SoNG (portalS of Next Generation), a joint development of 13 European academic and industrial partners in the framework of the same named IST project. An example of application of the presented technology and of its integration with autonomous agents is given: embodied virtual assistants interacting with human users. A stronger emphasis is put on the body animation aspects.

3.2 Face and Body Animation in MPEG-4

3.2.1 Overview

MPEG-4 Face and Body technology[2] primarily addresses two types of requirements: (1) it provides a way to specify the geometry of virtual humans' 3D faces and bodies, (2) it defines sets of standardized animation parameters that can be safely applied to the provided geometry. In addition, the issue of the efficient compression and streaming of face and body data is addressed by the standard.

MPEG-4 compliant faces are defined as polygonal mesh-based representations; bodies are composed of a hierarchically organised set of joints and attached segments. In order to be displayed and animated, the face and body geometry needs to be described in a specific format. Support for Face and Body Animation (FBA) in a MPEG-4 compliant system is handled by means of so-called "BIFS nodes", referenced in the scene description, that are animated using the output of a dedicated FBA decoder. BIFS, an abbreviation for "BInary Format for Scenes", provides a framework for the description of complex scenes composed of 2D and 3D graphics elements. The MPEG-4 specification defines a number of BIFS nodes inherited from the VRML 2.0 standard (*VRML homepage)*, and other specific nodes like *Face* and *Body* which allow for the efficient coding/decoding of face and body geometry. A description of these nodes is given in Appendix.

Standardized animation parameters play a central part in MPEG-4 face and body animation. For facial animation, MPEG-4 defines certain key locations on the face as feature points (see Figure 6). The Facial Animation Parameters (FAPs) are defined in terms of the normalized displacements of these feature points from their neutral positions, thus resulting in different

[2] MPEG-4 is a rich and complex standard which incorporates very different aspects in addition to face and body animation, such as the streaming of audio-visual content or the encoding of generic 3D geometry. Although we try to describe face and body animation in MPEG-4 independently from the other parts of the standard, it may be necessary to check additional resources for a comprehensive view of the full scale MPEG-4 architecture and of the related integration issues. More detailed descriptions can be found in the standard (*ISO/IEC JTC 1/SC 29/WG11 N2502* 1998; *ISO/IEC JTC 1/SC 29/WG11 N2739 subpart 2* 1999).

expressions. There are 66 low-level FAPs and 2 high-level FAPs. Stretch right corner lip, raise left inner eyebrow, puff cheek, etc. are examples of low-level FAPs. A sample of the standardized FAPs can be found in Table 2.

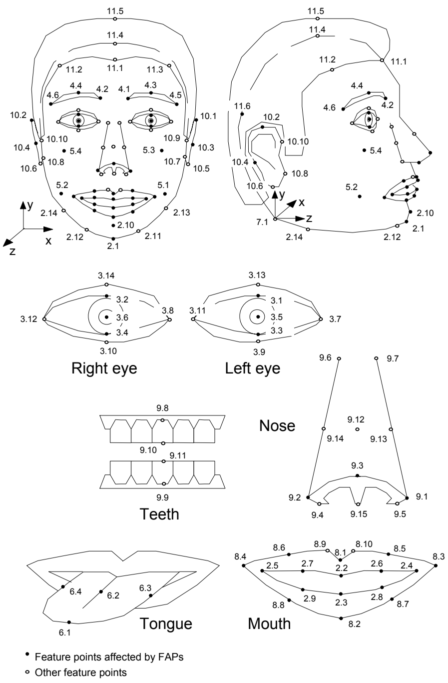


Figure 6 Facial feature points

#	FAP name	FAP description
1	Viseme	Set of values determining the mixture of two visemes for this frame
2	Expression	A set of values determining the mixture of two facial expression
3	open_jaw	Vertical jaw displacement (does not affect mouth opening)
4	lower_t_midlip	Vertical top middle inner lip displacement
5	raise_b_midlip	Vertical bottom middle inner lip displacement
6	stretch_l_cornerlip	Horizontal displacement of left inner lip corner
7	stretch_r_cornerlip	Horizontal displacement of right inner lip corner
8	lower_t_lip_lm	Vertical displacement of midpoint between left corner and middle of top inner lip
9	lower_t_lip_rm	Vertical displacement of midpoint between right corner and middle of top inner lip
10	raise_b_lip_lm	Vertical displacement of midpoint between left corner and middle of bottom inner lip

Table 2 A sample of FAPs

The two high-level FAPs correspond to visemes and expressions. A viseme is a visual correlate to a phoneme. Each viseme can take one of the 14 pre-defined values corresponding to groups of phonemes. Expressions are particularly important for the simulation of face-to-face interaction. Six high-level expressions are readily available in MPEG-4:

- **Joy**: The eyebrows are relaxed. The mouth is open and the mouth corners pulled back toward the ears.
- **Sadness**: The inner eyebrows are bent upward. The eyes are slightly closed. The mouth is relaxed.
- **Anger**: The inner eyebrows are pulled downward and together. The eyes are wide open. The lips are pressed against each other or opened to expose the teeth.
- **Disgust**: The eyebrows are raised and pulled together. The inner eyebrows are bent upward. The eyes are tense and alert.
- **Fear**: The eyebrows and eyelids are relaxed. The upper lip is raised and curled, often asymmetrically.
- **Surprise**: The eyebrows are raised. The upper eyelids are wide open, the lower relaxed. The jaw is opened.

Depending on the application, the high-level parameters can be specified in terms of low-level parameters when precision and variety are of prime importance.

Body animation in MPEG-4 is defined in terms of 296 Body Animation Parameters (BAPs), which represent a certain Degree of Freedom (DOF) of a given body articulation standardized in the H-Anim standard, e.g. sacroiliac tilt, left hip twisting, skull base rotation, etc. All BAPs together define a specific state of the human skeleton. Degrees of freedom can be translations or rotations along an axis, and the corresponding BAP values are stored as encoded integer values. Here is how we compute rotational BAP values. Given:

- a rotation angle in radian of a joint's degree of freedom $DOFRadAngle \in \mathbf{R}$

$$BAP_{DOF} = \frac{DOFRandAngle * 100000}{PI}$$

Translation are directly stocked with a precision of one centimeter. Given:

- a translation value in meters of a joint's degree of freedom $DOFDistInMeters \in \mathbf{R}$

$$BAP_{DOF} = DOFDistInMeters * 100$$

A full body animation requires 186 basic BAPs; 110 other BAPs being reserved for extensions like tail animation or to deform some parts of the body. In addition, the BAPs have been classified in a number of groups (e.g. Pelvis, Left leg1, etc.), which allows a more efficient update of the joints. A sample of the standardized BAPs is given in Table 3.

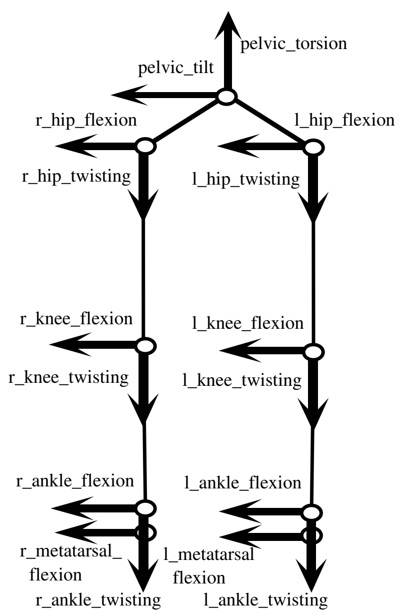


Figure 7 A sample of the body topology

#	BAP name	BAP description
1	sacroiliac_tilt	Forward-backward motion of the pelvis in the sagittal plane
2	sacroiliac_torsion	Rotation of the pelvis along the body vertical axis (defined by skeleton root)
3	sacroiliac_roll	Side to side swinging of the pelvis in the coronal plane
4	l_hip_flexion	Forward-backward rotation in the sagittal plane
5	r_hip_flexion	Forward-backward rotation in the sagittal plane
6	l_hip_abduct	Sideward opening in the coronal plane
7	r_hip_abduct	Sideward opening in the coronal plane
8	l_hip_twisting	Rotation along the thigh axis
9	r_hip_twisting	Rotation along the thigh axis
10	l_knee_flexion	Flexion-extension of the leg in the sagittal plane

Table 3 A sample of BAPs

3.2.2 Implementation of MPEG-4 Body Animation Components

In this section, we describe the components we have implemented in order to provide compliant body animation to MPEG-4 systems. We focus on animation technique itself and let aside the parsing and rendering of the BIFS nodes. The main problem, after having obtained a stream of BAP values (from a file, a specific decoder, etc.), is to create a system that efficiently interprets the BAPs as geometric transformations and applies these transformations in the body sub-scene graph parts that represent articulations. This is the most important task that the components described in this section achieve.

3.2.2.1 MPEG4SkeletonLib

MPEG4SkeletonLib is a C++ library that handles virtual (in the sense of non-visual) MPEG-4 compliant skeleton structures. It provides the following features:

- animation of a scene graph representing a body using BAPs,
- real-time execution,
- simultaneous animation of multiple bodies,
- high scene graph implementation independence,
- full MPEG-4 body animation compliancy

The library is based on four main classes: DegreeOfFreedom, Joint, JointAdapter and MPEG4InternalSkeleton. DegreeOfFreedom works as an abstraction for BAPs, which can be assigned to it through the interface it provides. Since each BAP has an effect on a specific joint (see Figure 8 for an example), which can be either a rotation on the X, Y or Z axis or a translation along the X, Y or Z axis, objects of the DegreeOfFreedom class contain a reference to the Joint they belong to. DegreeOfFreedom has been derived into 6 subclasses, one for each axis and each type of action (3 axis and 2 actions, rotation or translation), that implement specific methods to update the corresponding joint matrix. Several mechanisms have been implemented in order to efficiently process a high as well as a limited number of BAP updates.

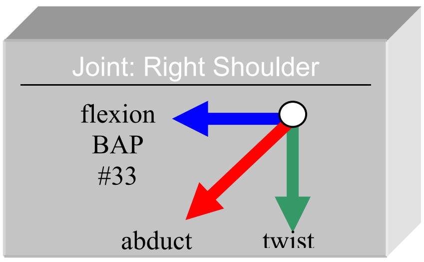


Figure 8 A BAP as a degree of freedom linked to a specific joint

A Joint object contains a reference to every DegreeOfFreedom it handles, as well as a specification of the order of rotation processing it must follow, as defined in the MPEG-4 standard. In addition, the 87 standard MPEG-4 Joints have their own referential in space, defined by a rotation of the global referential. We call "*toStandard*" the transformation that leads to the local Joint referential, starting from the standard world referential. In the example given in Figure 9, this transformation is a simple rotation of -Π/2 along the Y axis. The

DegreeOfFreedom associated with the abduction is the X rotation (red) on that referential, the flexion is the Z rotation (blue) and finally the twist is the Y rotation (green).

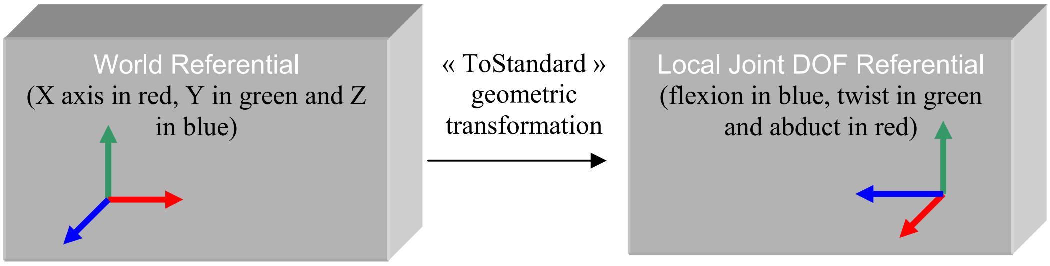


Figure 9 From world referential to MPEG-4 standard Joint referential

A Joint can be seen as a collection of local degrees of freedom transformations defined by BAPs. However, in order to update the scene graph, we must calculate the resulting transformation in the world referential. Thus, we have to pre-multiply this local transformation by the *toStandard* matrix, then apply the obtained transformation, and then go back to the world referential by post-multiplying by the inverse of the *toStandard* matrix. Given:

- the result matrix in world referential *JointWorldMatrix*,
- the transformation leading to the local Joint referential *toStandard*,
- the local joint transformation *Local*

$$JointWorldMatrix = toStandard * Local * toStandard^{-1}$$

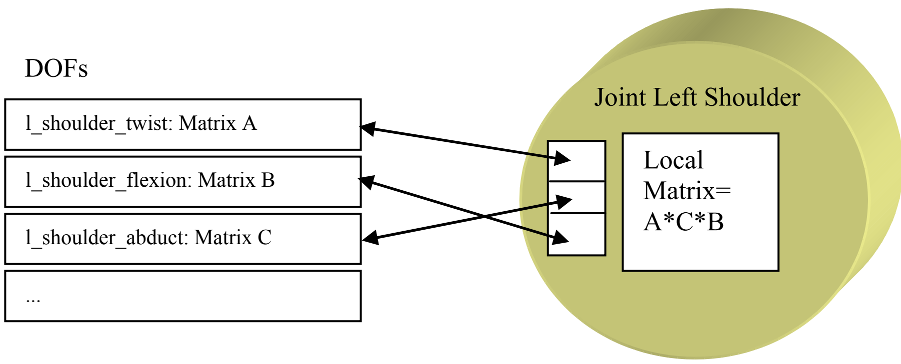


Figure 10 The local Joint matrix computation

The JointAdapter class is used to connect the joints of our virtual skeleton to the nodes of a specific scene graph. When a joint has been updated by a frame of BAPs and the transformation matrix has been computed, the corresponding scene graph *Transform* node must be updated by calling some dedicated scene graph functions. The advantage of using a separate JointAdapter class is that, when a major change occurs in the scene graph implementation, no recompilation of MPEG4SkeletonLib is required. Joint objects call virtual methods of their assigned JointAdapter without knowing anything from the real object that will be linked to it. On the scene graph side, a class derived from JointAdapter implements these methods. Taking advantage of C++ polymorphism, we have an interface on the joint side, that is at runtime automatically redirected to a correct implementation on the scene graph side.

Finally, the MPEG4InternalSkeleton class serves as a global container for Joint and DegreeOfFreedom objects. When instantiated, it creates the full set of MPEG-4 joints and links them hierarchically. The DegreeOfFredom objects are built at the same time and the already described bi-directional link with the Joint objects is established. This structure, once linked to a scene graph, is ready to be animated by MPEG-4 Body Animation Parameters.

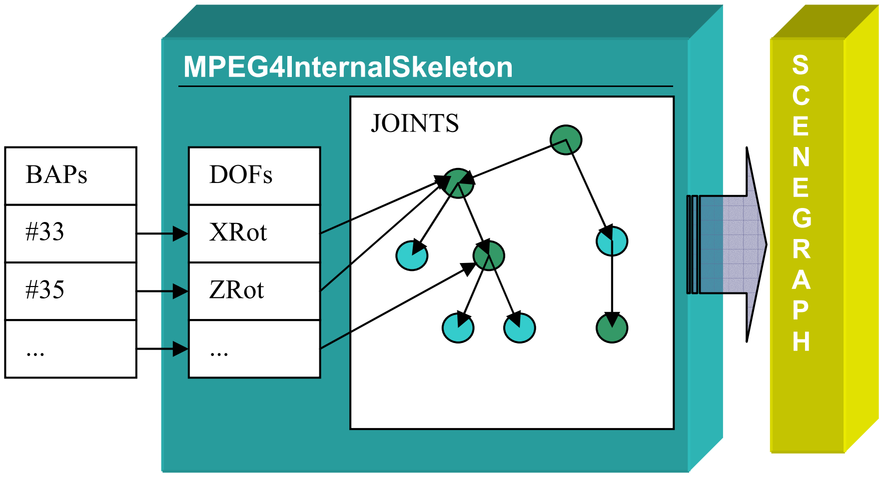


Figure 11 From BAPs to scenegraph updates

3.2.2.2 MPEG4AnimationLib

MPEG4AnimationLib is a C++ library devoted to the management of MPEG-4 Body Animation Parameters. It can load and save BAP frames from/to .bap files, a simple text file format (see Figure 12 for an example). In addition to BAP values (integer) and BAP Mask values (boolean), important information like the number of frames and the frame rate is included, and each frame of an animation can be accessed independently. MPEG4AnimationLib provides the input to MPEG4SkeletonLib in the correct format.

Using the library, animations can be processed in progressive or absolute mode.

- In progressive mode (the default mode), animations consist of BAP values that are modified in each frame. The BAP Mask is used to indicate the BAPs that must be updated. This mode does not allow random access to specific frames, because every frame is dependent on the former ones, but is very efficient both in terms of processing speed and required memory.
- In absolute mode, each frame is considered as a full set of BAPs to be applied to a body. This allows random access to single animation frames but leads to an overhead in the number of unnecessary BAPs that are applied.

```
# BAP file v3.2, EPFL-VRLAB, Switzerland, © 1999 - 2003
# Header description: version, filename, frame rate, number of frames
# BAP mask: one 0 or 1 for each of the possible 296 BAPs (masked or unmasked) indicating
# whether it is used. The BAP mask precedes each BAP data line on a separate line.
# Each frame is described as one line of integer BAP values starting with the
# frame number. A zero value for any BAP indicates a body in default position.
3.2 charleston.TRK 25 231
0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 1 0 0 1 1 1 1 0 0 1 1 0 1 0 0 1 1 1 1 0 0 1 1 0 0 0 0 1 1 1 1 0 0 1 1 0 0 0 0 1 1 1 1 0 0 1 1 1 0 0 0 1 1 1 1 1 1 1 1
1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 19299 7651 2478 2710 -317 -2744 -23857 -14047 -2955 3154 -1806 4847 8129 -7366 -15534 5547 51281
41498 14729 670 15690 28437 -1164 6977 91 14090 -116 807 -3804 7241 -4065 -6693 6641 -7803 -4614 -19 -
2808 -9923 -3402 -177 -7803 -7009 4585 -6450 -101 1752 -12992 -4580 317 -6675 3600 -3383 -10555 -7517 -
2413 -12047 2442 17749 -1838 -11656 -5862 -12977 -13645 -4825 -2356 -100000 6087
...
```

Figure 12 A sample of .bap file

3.2.2.3 SimpleBAPViewer

In order to be able to conveniently test our libraries and animations independently from the complexity inherent to a full MPEG-4 system, we have implemented a simple OpenGL BAP viewer shown in Figure 13. It is linked with MPEG4AnimationLib and MPEG4SkeletonLib, and animates a MPEG-4 skeleton embodiment using .bap or .fba (see Section 3.2.2.5) files.

3.2.2.4 BAP Production Plug-in

The previously described MPEG4SkeletonLib library is also used for the production of MPEG-4 animations.

In order to constitute a large database of .bap files, we have developed a 3DS Max exporter plug-in that converts, in the BAP format, animations conveniently designed using the Bones System user interface. As in a player implementation, the role of MPEG4SkeletonLib is to properly connect both scene graphs (MPEG4SkeletonLib's virtual skeleton joints structure and 3DS Max's hierarchy of INode objects) for the extraction of adequate rotation and translation values.

The first step is to perform a recursive search of each joint's name using 3DS Max SDK. Then, for each frame, the found joint's local transformations are retrieved, converted to the correct referential and decomposed into BAP values. The involved complexity is here somewhat higher than in a player implementation: since all joints' transformations are local to their parent joints', the corresponding matrices must be recursively multiplied starting from the root node.

The JointAdapter model used in MPEG4SkeletonLib allows the library to connect as easily to an OpenGL scene graph in order to play body animations (like described in the preceding section), as to a proprietary framework such as 3DS Max for production purposes.

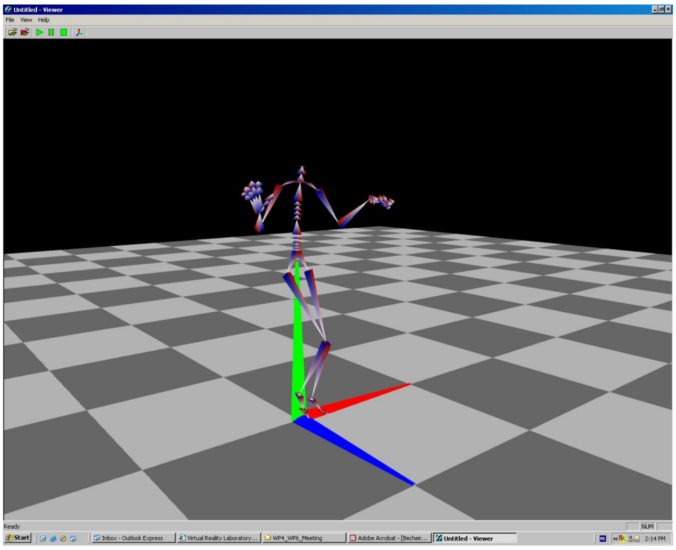


Figure 13 The SimpleBAPViewer

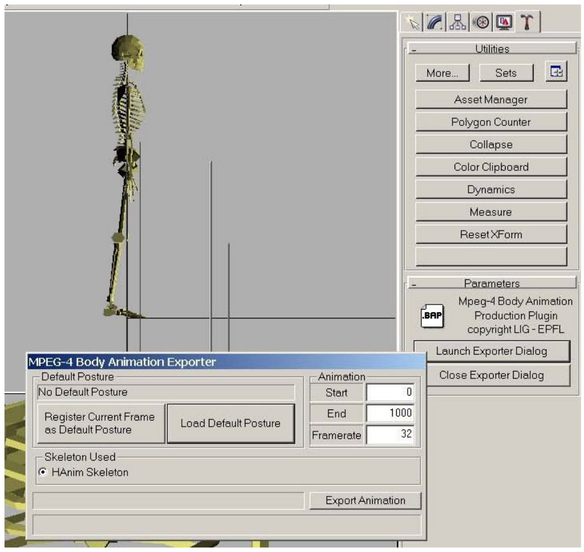


Figure 14 The BAP production plug-in

3.2.2.5 FBA Stream Encoder and Decoder

According to the MPEG-4 specification, FAPs and BAPs must be encoded together in a compressed FBA stream. A quick overview is given in this section; the precise format of the bitstream as well as encoding and decoding algorithms are detailed in the standard (*ISO/IEC JTC 1/SC 29/WG11 N2739 subpart 2* 1999).

The FBA stream is subdivided into Access Units (AU), each associated with a BIFS-Anim compliant header. The BIFS-Anim stream encapsulates FBA data and provides, through specific ObjectDescriptors, a link to the face and body nodes that must be animated.

The header of the FBA stream contains a 2 bits mask that indicates whether or not the stream contains facial and body animation. This enables the system to animate a face without body (e.g. for 3D conferencing applications), a body without an animated face (e.g. for low-performance setups) and face and body together (e.g. to realistically simulate embodied social interaction). The AU headers contain optimization information, like a full FAP mask or the IDs of updated BAP groups.

FAPs and BAPs encoding fits the general MPEG approach and relies on predictive coding and quantization techniques. Some frames of the animation are encoded in Intra mode (I-frames), i.e. the parameters values are directly quantized. Others are stored as predictive frames (P-frames), i.e. the delta changes with the previous frame's values are quantized. The quantization step determines the quality of the animation, i.e. the more quantized a stream, the lower the size of the stream and the quality of the decoded data. On the contrary, fine quantization will lead to better quality animation but also to a larger bitstream. A range estimate for each parameter is also provided to the encoder, which improves the coding efficiency. This scheme has the advantage of preventing the accumulation of prediction errors. We did not consider the DCT-based approach which is another coding scheme specified in the standard.

We have developed a FBA encoder which generates compliant bitstreams starting from BAP frames, and a FBA decoder which is able to parse FBA streams and decode BAP values in real-time. They are based on an arithmetic encoder/decoder coupled to a circular buffer. On top of it, producer and consumer threads, synchronized using semaphors, manage the input and the output of the BAP/FBA data.

As dynamic libraries (DLL), they can be easily integrated in any MPEG-4 player or desktop tool (see Figure 15). However, the encapsulation of the resulting data in a BIFS-Anim stream requires additional MPEG-4 software (e.g. MP4Tool).

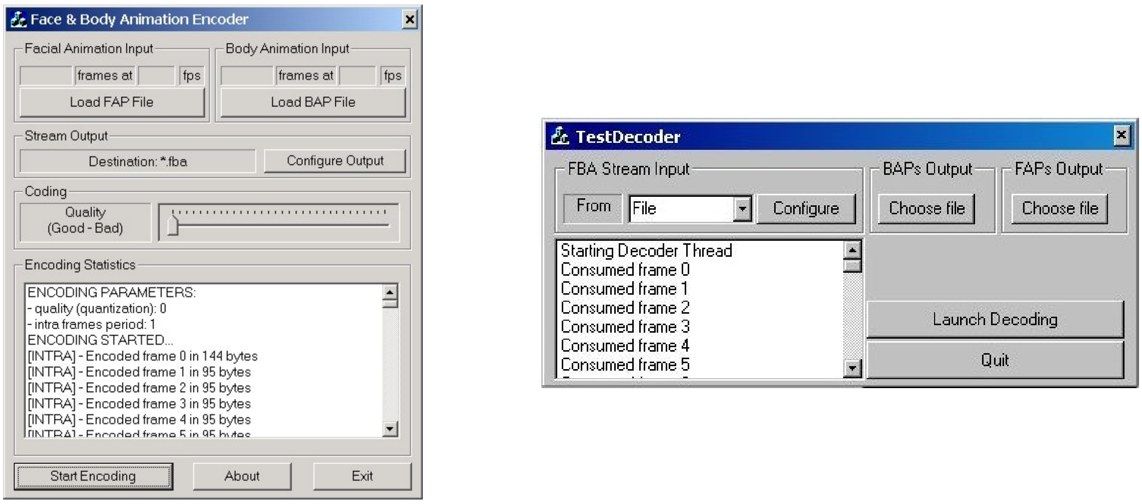


Figure 15 FBA encoder and decoder tools

3.3 Avatar Markup Language

Beyond the low level services provided by the components we have just presented, synchronization of speech, facial expressions and body movements is a critical problem in the simulation of embodied social interaction. In this section, we address this problem by proposing a new high-level animation language to describe the avatar animation. The Avatar Markup Language - AML (Kshirsagar et al. 2002; Guye-Vuillème & Thalmann 2002; Arafa et al. 2002), based on XML, encapsulates Text To Speech (TTS), facial animation and body animation in a unified manner with appropriate synchronization. It enables easy and rapid design of avatar animations by simulation designers, without requiring intricate knowledge of the low-level parameters controlling the 3D avatar representation.

In the context of simulation systems where embodiments are controlled by autonomous agents, what is missing in order to produce rich nonverbal interactions is an intermediate layer between the decision making ability of the agent and the low-level rendering capabilities of, for example, a MPEG-4 player. Although it would be possible for the agent to directly generate the low-level animation data, it would place a useless burden on it and would make difficult for simulation designers to finely control the behavioural outcome.

For example, in an interaction where a shop assistant welcomes a virtual customer, it would smile, bow a little, and say, "May I help you?". In order to facilitate such multi-modality in its interactions, the agent is required to trigger the appropriate TTS, face animation and body animation modules in a time-synchronized and easy manner. This may involve mixing multiple gestures and expressions into a single animation. At the same time, the context in which the agent must operate varies and may be a function of a user's behaviour, its own previous actions, events in the environment, etc. It is therefore necessary to allow the agent to trigger parametric behaviours such as pointing at an object or moving to a specific coordinate in the 3D space.

Various scenarios were identified, in which such unified synchronized face and body animations with TTS output may be required. Figure 16 shows three tracks, one each for face animation, body animation and TTS. A red line indicates the presence of the particular track during that time interval. It is important to consider the situations where face animation and TTS overlap. For example, the avatar may be smiling and starts speaking as it continues to smile. Also, appropriate co-articulation needs to be implemented while rendering realistic speech animation. The carefully weighted sum of facial animation parameters is required in order to avoid jerky animation or artefacts (Kshirsagar, Molet & Magnenat-Thalmann 2001). In addition, in order to have proper body animation synchronized either with speech or with facial expressions, appropriate positioning of these tracks is necessary. The complex scenarios with overlapping animation elements, as described above, must be supported in order to allow flexible and seamless animation.

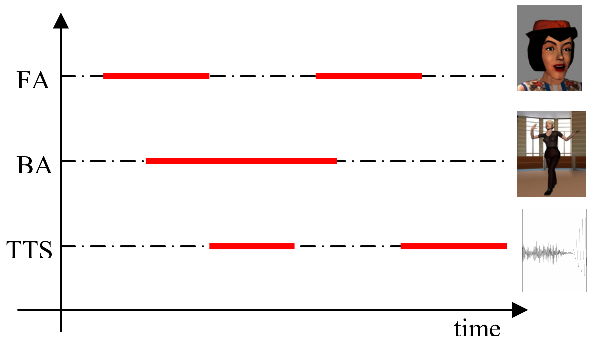


Figure 16 Animation scenario

We argue that a high-level definition language is necessary to assemble the interaction modalities into a unified animation while providing the required flexibility. The added value of an extra layer allowing the reuse of predefined animations by providing parametrization facilities, is also to be taken into account.

After a short discussion of related work, we discuss in more details the requirements, and present the AML syntax as well as our reference implementation based on MPEG-4 animation parameters.

3.3.1 Related Work

Scripting systems have always been a very popular motion control technique in computer animation. Nevertheless, relatively few script-based systems have been developed in order to provide high-level control mechanisms for character animation. Using a rule-based approach, IMPROV (Perlin & Goldberg 1996) allows designers to write sophisticated scripts triggering the generation of smooth and non-repetitive body motions in real-time. BEAT and BodyChat (Cassell, Vilhjalmsson & Bickmore 2001) automatically trigger body behaviours based on a textual input, using a set of rules that can be modified without recompilation. These advanced systems do not allow a clear separation between the purely deliberating function and the behaviour specification itself.

The Motivate system (*Motion Factory homepage*) is another example of a solution which uses high-level scripts in order to model virtual humans' behaviour. It is based on a hierarchical finite state machine and is mainly aimed at the gaming industry. Developed at EPFL, ACE (Kallmann et al. 2000), a Python-based interpreter built on top of several animation and perception libraries, offers much flexibility to advanced designers who can build powerful behavioural scripts for intelligent agents. Nevertheless, like several other interesting systems that could have been cited here, ACE is based on a proprietary animation technology and is, at least to some extent, application specific.

Recently, a number of XML-based animation languages have appeared. The Virtual Human Markup Language (VHML) (*VHML homepage*) is a system independent XML-based language which can be used to trigger facial and body animation easily. Though, in its current state of development, it is missing any time synchronization information, which is crucial to seamless animation. Moreover, it doesn't handle any reference to the environment (e.g. "look_left", "look_right" tags can't be used to direct the avatar's attention to a precise location in the scene, an object, etc.), which makes it more suited to nonverbal chat scenarios ("talking head/trunk" type) than to applications where embodied agents are immersed in rich shared virtual environments. Two other interesting XML languages have to be cited: the Character Markup Language (CML) (Arafa & Mamdani 2003) - a comparative study with AML can be found in one of our publications (Arafa et al. 2002), and the high-level Affective Presentation Markup Language (APML) (De Carolis et al. 2002), developed at the University of Rome. The emergence of these languages demonstrates that the need that AML aims to satisfy, is widely shared within the field.

3.3.2 AML Requirements

The analysis of various scenarios and use cases, such as the one given in introduction, led to the identification and formalization of several requirements which are presented in this section.

3.3.2.1 Predefined Face and Body Animations

The ability for AML documents to reference reusable predefined animations is crucial. Accordingly, databases of facial expressions (FAP DB) and body animations (BAP DB) must be provided with the parser. The content of these databases can be partly standardized for common nonverbal interactions (e.g. chat), another part being inevitably application specific. All the definition files for the high level FAPs must be included, as well as a number of additional “non-standard” expression files useful to add variety. In the BAP DB, it is important to have gestures, i.e. animations sequences returning to their initial states, as well as postures, which result in a new final skeleton configuration. Physically, in a MPEG-4 implementation, the databases are constituted of .ex (standard facial expressions), .fap and .bap files.

Figure 17 Face and body animation snapshots from the FAP and BAP databases

3.3.2.2 Organization of Animations in the Temporal Dimension

It is a basic requirement to be able to provide a description of the agent behaviour in the temporal dimension. The number of referenced animations should not be artificially limited, and designers should be able to assign an individual starting time to each of them. We have chosen a time format relative to the beginning of the AML animation, based on minutes, seconds and milliseconds (mm:ss.mmm). As previously mentioned, the generation of a continuous signal cannot be presupposed, e.g. an avatar should be able to adopt a posture, be inactive for 2 seconds, and resume its activity.

In order to permit the specification of parallel behaviours, overlapping sequences should be allowed. In this case, the parser should be able to automatically perform motion blending so that it can still generate a smooth animation. A priority mechanism may be introduced, allowing designers to decide which animation should be used when the same parameter is referenced or whether an intermediate skeleton configuration must be interpolated.

Since the duration of parametric actions may not be known at design time, a synchronization mode has to be proposed as an alternative to the starting time parameter, which allows actions to be triggered relative to each other, e.g. at the same time as the preceding action or directly after the preceding action has completed. There should be no limitations in combining animations triggered using a start time or a synchronization mode.

3.3.2.3 Synchronization with Speech

A Text To Speech interface allowing the integration of speech and its synchronization with lips movements, must be provided in the language. The TTS engine must receive text as input from the AML Processor, and output a combination of an audio stream (output from a TTS synthesizer) and a phoneme sequence with appropriate timing information, to be used by the facial animation engine.

Another desirable feature is to be able to annotate the text with tags, in order to embed emotions. These tags should affect the audio as well as the resulting FAPs, so that as a result, the avatar sounds happy and looks happy, for example. We do not specify a format, thus syntaxes such as Microsoft SAPI tags (*Microsoft Speech Technologies homepage*) or MPEG-4 defined TTS bookmarks (*ISO/IEC JTC 1/SC 29/WG11 N2502* 1998) can be used depending on the implementation.

3.3.2.4 Real-time Customisation of Predefined Animations

In order to facilitate reuse and finely adapt predefined animations to the context, it is important to provide real-time customisation facilities. The envelope of face and body animations should be adjustable to some extent. For facial animations, a normalized intensity setting should be provided to control the maximum displacements of the feature points. A way to control the expression duration should also exist. For body animations, the speed of the animation is the first required parameter, e.g. a fast (enthusiastic) head nodding is not interpreted in the same way as its slow motion equivalent (reluctant). In some cases, it is also important to allow changes in the gestures' amplitude by increasing/decreasing joint angles. This must be possible by simply modifying a body animation intensity setting.

So that animations can be further reused and the parser's performance improved, the use of masks to inhibit the animation of some parts of the face/skeleton is valuable. For example, if a full body animation is to be triggered while a walking action is performed, designers should be able to easily deactivate lower body updates. For facial animation in MPEG-4 systems, the mask is composed of 68 boolean values, each of them linked to one specific FAP. 296 values are necessary for a MPEG-4 BAP mask. Predefined high-level masks such as “lower body”, “right arm”, etc. may also be provided to facilitate the operation.

3.3.2.5 Real-time Parametric Animations

There are body actions that cannot be pre-generated because they are dependant on dynamic and unpredictable conditions, like the position or orientation of the avatar in the scene. Some behaviours, e.g. pointing or walking to specific coordinates in 3D space, must then be generated on-the-fly using appropriate techniques, from a set of specific parameters given as input in the script. For the initial implementation, we restricted the list of parametric animations to the following actions: facing, pointing, walking, resetting and waiting.

In order to generate correct parametric animations and smooth transitions between the processing of different AML documents, it is important that the parser maintains a set of critical information, like the avatar position in the scene, the body global orientation, and the skeleton configuration. In MPEG-4 systems, the parser must keep running between calls and can simply save the full array of generated/blended FAPs and BAPs at each frame. It is also necessary to have a way of providing additional information to the parser, like body measurements, so that it can generate adequate motions (e.g. translation to the right shoulder origin for a pointing behaviour).

3.3.3 AML Syntax

Figure 18 shows the AML syntax with the basic elements. The root tag *<AML>* marks the beginning and end of the script. It accepts four attributes: *face_id* (the reference id for the 3D face to be animated), *body_id* (the reference id for the 3D body to be animated), *root_path* (the root path for animation files such as expression files, FAPs, and BAPs, explained in detail in the next subsections) and *name* (the name of the animation project). *<AML>* has two sub-elements: *<FA>* or Facial Animation and *<BA>* or Body Animation.

The *<FA>* and *<BA>* tags both accept two attributes: *start_time* (the relative start time of the respective scripts) and *input_file* (the name of an Avatar Face Markup Language (AFML) or

Avatar Body Markup Language (ABML) file if they are predefined; *input_file* can also have the value “none”). In the case that *input_file* is “*none*”, the AFML and ABML are defined within the AML script.

```
<AML face_id =“x” body_id = “y” root_path = “p” name = “name of project”>
<FA start_time = “t1” input_file = “f1”>
<TTS mode = “m” start_time = “t3” output_fap = “f3” output_wav = “f4”>
<Text>TextToBeSpoken<\Text>
<\TTS>
<AFML>…<\AFML>
<\FA>
<BA start_time = “t2” input_file = “f2”>
<ABML>…<\ABML>
<\BA>
<\AML>
```

Figure 18 AML syntax

The <*FA*> further has an optional sub-element <*TTS*> or Text To Speech. The <*TTS*> tag accepts four attributes: mode (“*annotated*” if the TTS input is annotated with emotional tags, “*plain*” otherwise), *start_time* (the relative start time of the TTS rendering), *output_fap* (the filename to use when generating FAP output) and *output_wav* (the filename to use when generating audio output). Though we use the .fap file as output here, it may be possible to use any other format (viseme or phoneme), provided the parser and further processing are designed accordingly. <*TTS*> has a sub-element <*Text*>, which defines the text to send to the TTS engine. The speech animation can be added either in <*TTS*> tag or directly in the AFML, as will be explained in the following sub-sections.

3.3.3.1 Avatar Face Markup Language (AFML)

This is a high-level description language for facial animation. We use MPEG-4 FAP as low-level parameters for animation. The FAP Database (FAP DB) contains a variety of pre-defined facial expressions, defined in terms of MPEG-4 FAPs. The <*Settings*> tag contains information like the frame-rate, the length of animation, and the local path of FAP DB as well as pre-defined speech animations, if any. A number of tracks can be defined in the AFML. Each track can be given a name; e.g. emotions, head movements, eye movements etc. This separation enables distinct control over various parts of the face. There may be as many <*ExpressionsTrack*> elements as required. Further, there may be as many <*Expression*> elements as required in each <*ExpressionsTrack*>. The expressions may or may not be overlapping. Each <*Expression*> has a start time, a name, and a time envelope defined. The name, in fact, refers to the static expression file from the FAP DB. Each time envelope is defined by as many <*Point*> elements as required. The shape defines the interpolation from the previous point to the current one, and the interpolation can be either logarithmic, exponential or linear. The first default point is with zero intensity at the start time. The intensity is normalized, and duration is specified in seconds. The <*SpeechTrack*> is reserved for the viseme or fap files corresponding to a pre-defined speech animation. The viseme file contains timing information for each of the visemes, and the fap file contains frame-by-frame information of the low-level facial animation parameters for the speech animation. The speech track also specifies the audio file for the pre-recorded speech by the <*AudioFile*> tag. It should be noted that the inclusion of the speech track enables the use of pre-defined or pre-recorded speech animations. Unlike the expression tracks, the speech tracks cannot be overlapping.

```
<AFML>
<Settings>
<Fps>FramesPerSecond</Fps>
<Duration>mm:ss:mmm</Duration>
<FAPDBPath>“path of folder containing high level expression (.ex) files”</FAPDBPath>
<SpeechPath>“path of folder containing speech animation (.vis) files”</SpeechPath>
</Settings>
<ExpressionsFiles>
<File>“name of expression file (.ex) from the path defined above”</File>
</ExpressionsFiles>
<ExpressionsTrack name=“Name of track”>
<Expression>
<StartTime>mm:ss:mmm</StartTime>
<ExpressionName>“name”</ExpressionName>
<Envelope>
<Point>
<Shape>{log or exp or linear}</Shape><Duration>InSeconds</Duration>
<Intensity>NormalizedIntensity</Intensity>
</Point>
</Envelope>
</Expression>
</ExpressionsTrack>
<SpeechTrack name=“name_of_track”>
<StartTime>mm:ss:mmm</StartTime>
<FileName>“viseme or fap file name”</FileName><AudioFile>“AudioFileName”</AudioFile>
</SpeechTrack>
</AFML>
```

Figure 19 AFML syntax

```
<ExpressionsTrack name=“Emotions”>
<Expression>
<StartTime>00:00:800</StartTime>
<ExpressionName>smile.ex</ExpressionName>
<Envelope>
<Point>
<Shape>log</Shape><Duration>0.5<Duration>
<Intensity>1<Intensity>
</Point>
<Point>
<Shape>log</Shape><Duration>0.5<Duration>
<Intensity>0.7<Intensity>
</Point>
<Point>
<Shape>exp</Shape><Duration>0.8<Duration>
<Intensity>0<Intensity>
</Point>
</Envelope>
</Expression>
</ExpressionsTrack>
```

Figure 20 An example of an expression track

```
<SpeechTrack>
<StartTime>00:07:600</StartTime>
<FileName>speech1.fap</FileName>
<AudioFile>speech1.wav</Audiofile>
</SpeechTrack>
```

Figure 21 An example of a speech track

3.3.3.2 Avatar Body Markup Language (ABML)

ABML is very similar in format to AFML. The <*Settings*> tag is the same except that it contains one <*BAPLibPath*>, to locate the pre-defined BAP animations, in addition to the <*Fps*> and <*Duration*> tags. There can be as many <*BodyAnimationTrack*> as needed, each specifying certain actions. The <*Mask*> tag can be used to indicate which BAPs are active for a particular track, so that the calculation of all the BAPs is not necessary. Since this tag is specific to MPEG-4 BAPs, it is optional. The body animation uses BAP files (using the <*PredefinedAnimation*> tag) and behaviours (using various "*Action*" tags, explained in the next subsection) as the basic elements for animation. One or more body animation tracks can be defined as shown in Figure 22. Each track has a start time, speed and priority of the action. The speed can be set to: *slow*, *normal*, or *fast*. The <*Priority*> tag specifies which BAPs are to be used when several sequences overlap. An intermediate solution is automatically computed when the given priorities are equal. The <*Intensity*> tag for the pre-defined animation can be used to exaggerate or scale down the effect of the pre-defined animation.

```
<BodyAnimationTrack name="Name of track">
<Mask>int[296], the BAP indices that are affected by this track are set to 1, the rest 0 – optional tag</Mask>
<PredefinedAnimation>
<StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<FileName>"filename.bap"</FileName>
<Speed>{normal or slow or fast}</Speed>
<Intensity>0 to fn</Intensity><Priority>0 to n</Priority>
</PredefinedAnimation>
<FacingAction><StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<XCoor>target's X coordinate in meters</XCoor>
<YCoor>target's Y coordinate in meters</YCoor><ZCoor>target's Z coordinate in meters</ZCoor>
<Speed>{normal or slow or fast}</Speed><Priority>0 to n</Priority>
</FacingAction>
<PointingAction><StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<XCoor>target's X coordinate in meters</XCoor>
<YCoor>target's Y coordinate in meters</YCoor><ZCoor>target's Z coordinate in meters</ZCoor>
<Speed>{normal or slow or fast}</Speed><Priority>0 to n</Priority>
</PointingAction>
<WalkingAction><StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<ControlPoint>
<XCoor>target's X coordinate in meters</XCoor><ZCoor>target's Z coordinate in meters</ZCoor>
</ControlPoint>
<Speed>{normal or slow or fast}</Speed><Priority>0 to n</Priority>
</WalkingAction>
<ResettingAction><StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<Speed>{normal or slow or fast}</Speed><Priority>0 to n</Priority>
</ResettingAction>
<WaitingAction><StartTime>{mm:ss:mmm or autosynch or autoafter}</StartTime>
<Duration>float, duration in seconds - 0 to fn</Duration>
</WaitingAction>
</BodyAnimationTrack>
```

Figure 22 ABML syntax

The references to parametric animations (e.g. *PointingAction*) contain target locations specified by x, y, and z coordinates *(*<*XCoor*>, <*YCoor*>, and <*ZCoor*> tags) in 3D space. Additionally, the walking behaviour can specify any number of control points through which the avatar must pass, by using one or more <*ControlPoint*> tags.

Since the duration of parametric actions may not be known at design time, a "synchro mode" has been added to the ABML as an alternative to the starting time parameter, which allows actions to be triggered relative to each other: "*autosynch*", in place of the start time, indicates that the action will be triggered at the same time as the preceding action, "*autoafter*", in place of the start time, indicates that the action will be triggered directly after the preceding action has finished. There are no limitations to the possible combinations of "*start time*" / "*autosynch*" / "*autoafter*".

```
<ABML>
<Settings><Fps>25</Fps><Duration>00:10:000</Duration><BALibPath>.\BAPFiles\</BALibPath></Settings>
<BodyAnimationTrack name="Dance">
<PredefinedAnimation><StartTime>00:00:000</StartTime> <FileName>charleston.bap</FileName>
<Speed>normal</Speed> <Intensity>1</Intensity> <Priority>1</Priority></PredefinedAnimation>
</BodyAnimationTrack>
<BodyAnimationTrack name="FaceAndPoint">
<FacingAction><StartTime>00:02:000</StartTime><XCoor>-10</XCoor><YCoor>-10</YCoor><ZCoor>-
10</ZCoor><Speed>normal</Speed> <Priority>2</Priority></FacingAction>
<PointingAction>
<StartTime>autoafter</StartTime><XCoor>45</XCoor><YCoor>10</YCoor><ZCoor>10</ZCoor>
<Speed>normal</Speed> <Priority>2</Priority></PointingAction>
</BodyAnimationTrack>
</ABML>
```

Figure 23 ABML example

3.3.4 AML System Architecture and Implementation

Figure 24 shows the system architecture of our AML processor implementation that can be integrated in a MPEG-4 compliant real-time animation platform. The following subsections describe the main modules.

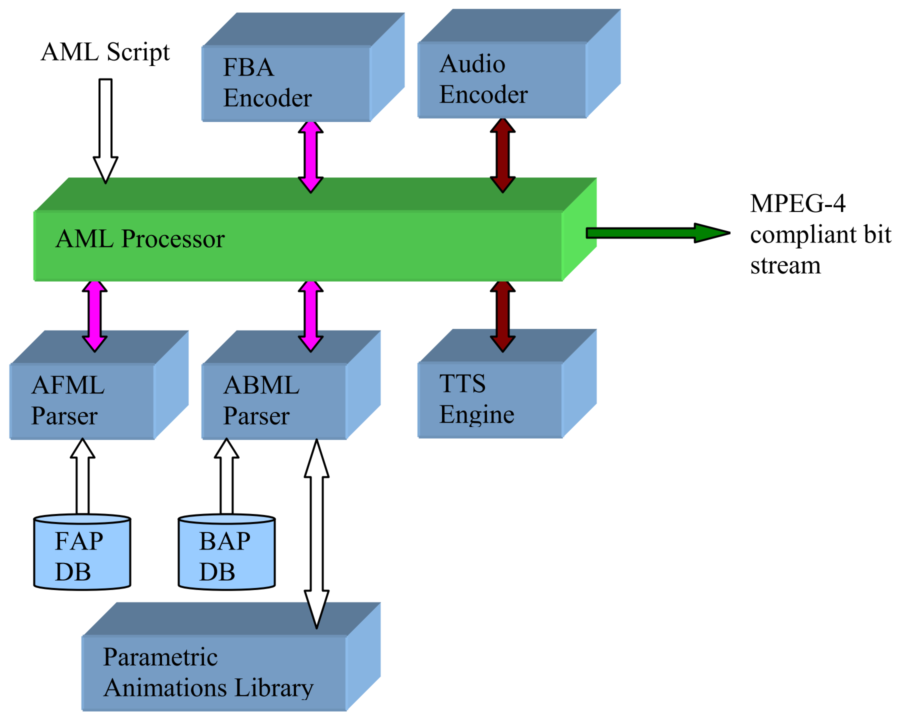


Figure 24 AML system architecture

3.3.4.1 AFML Parser

The AFML parser receives AFML as input, and can generate FAP streams by merging a number of expression tracks. Expression tracks are based on predefined animations stored in the FAP database, but parameters such as duration, intensity and time envelopes, can be varied using the tags of the AFML. AFML can also include a sequence of phonemes (a .vis file generated by TTS), which can be converted into lips animation by the AFML parser. Co-articulation for speech animation, blending between various expressions activated at a same time, and mixing of speech and expressions, are the important tasks of the AFML parser in order to generate smooth animations. The parser can read the AFML from either a file or a text stream, and can either generate a FAP file to be used for animation later, or return FAPs frame by frame when requested.

3.3.4.2 ABML Parser

Similar to AFML, ABML is parsed by a ABML parser, which generates BAP streams. The ABML parser can work with a number of body animation tracks containing predefined BAP files stored in the BAP DB, and it can generate body animations from parametric animation calls. The parser provides a way to organize bodily behaviours (gestures and postures) using timing information relative to each other (e.g. simultaneous or temporally serial), and allows the real-time modification of the animation's speed, intensity and extent (using masks). It is also able to compute intermediate skeleton states for several gestures (blending) when overlapping specifications are given, which users can control by assigning different priorities to each action.

Figure 25 AML blending of two dances: charleston and disco

The ABMLParser library has been developed in C++, intensively uses the Standard Template Library (STL), and takes advantage of powerful object-oriented features like polymorphism. Two main classes serve as a basis for the library: BASBody, which directs the whole process at a high-level and is in charge of maintaining information about the current body state, and BASScript, which parses the scripts, handles all references to .bap files and dynamic behaviours, retrieves and blends the adequate BAP frames in an orderly fashion. Blending is based on a simple weighted averaging approach, maintaining a dilution factor for each BAP. When .bap files are progressive (which is the case in our BAP DB), i.e. as explained in the previous section, it means that only the BAP updates are given, the signal must first be made continuous in order for the blending to give acceptable results. BAPFrame and its associated BAPMask class are used in most other classes. They efficiently handle one frame of indexed

BAP data, and provide some important low-level methods, such as computing masks union/intersection, averaging two frames of BAPs, etc.

BAPSource is the base class for every class providing BAPs to the parser, be it by reading and processing data from a file - BAPFile, or from an algorithmic source – SAFace, SAPoint, SAWalkTo, etc. These three specialized action classes are derived from a BAPGenerator class, which defines a set of matrix manipulation methods widely used in their implementation, e.g. quaternion interpolation. This architecture allows most of the code to be located in the base classes, thus keeping the derived classes very short and simple, which facilitates the addition of new behaviours. Moreover, the parsing and setup of the system become straightforward: each time a reference to a .bap file or to a behaviour is met, a corresponding C++ object is instantiated and placed in a STL vector of BAPSources. Then, at each frame and for every object in the vector, the parser calls the GetNextFrame method, which is dynamically bound to the adequate implementation depending on the type of BAP source. The algorithm is given in Figure 26.

```
While(at least one active BAPSource){
        BAPFrame resultframe;
        For each BAPSource in vector {
                If(BAPSource active){
                        BAPFrame tempframe=GetNextFrame();
                        MakeContinuousSignal(frametemp);
                        resultframe=resultframe+tempframe;
                }
        }
        MakeProgessiveSignal(resultframe);
        Send(resultframe);
}
```

Figure 26 ABML parser main algorithm

3.3.4.3 Parametric Animations Library

The Parametric Animations Library is a plugable dynamic library (DLL), which is in charge of generating on-the-fly BAPs and which is triggered by the ABML parser. Its design makes it very easy to add new behaviours and greatly facilitates the reuse of already developed parametric animations. The animations generated by the library can be processed (e.g. mixed with predefined animations, made faster/slower, etc.) transparently by the ABML parser, like any other predefined animation.

It is a C++ library which makes use of classic animation techniques based on matrix manipulation, like quaternion interpolation, and of inverse kinematics. IKAN (Tolani, Goswami & Badler 2000), an inverse kinematics library developed at the University of Pennsylvania, is used to compute, among other things, the final orientation of the arm in the Pointing action, given an elbow flexion parameter. It is also ideally suited to the implementation of new specialized actions like Touching or Grasping.

BAPGenerator and its derived classes are the main components of the Parametric Animations Library. Since they have already been presented, and in order to exemplify the flexible nature of the architecture, we now provide a short example explaining how a simple walk engine (SAWalkTo) has been implemented.

The first thing to consider is that most walk engines are based on a combination of translations and postural shifts. Besides, it should be mentioned that every BAPSource holds a pointer to its containing vector, which allows it to trigger other actions and control their execution. Given these two points, the subsequent steps can be followed in order to implement SAWalkTo:

1. At the beginning of the animation, a new SAFace action is to be triggered using the same timing and target parameters as the walk action. It will gradually modify the body and head orientations so that the target location is faced after the necessary number of frames.
2. Simultaneously, a predefined animation representing a short cyclic walking sequence (2 steps, typically generated from motion capture) should be run. At each frame, the BAPFile state has to be checked and a new one instantiated when the former is completed.
3. Simple translations have to be generated in the GetNextFrame method of the SAWalkTo object. When the target is reached, all child actions have to be marked as inactive and the posture should be reset by adding a final SAReset object to the BAPSource vector.

By varying the walking sequence used, different walking styles can also be generated.

Since all actions are processed in one single pipeline and every frame is automatically blended by the parser, such a simple procedure gives very good results in our architecture. As can be seen, the chosen approach is powerful and greatly facilitates code reuse.

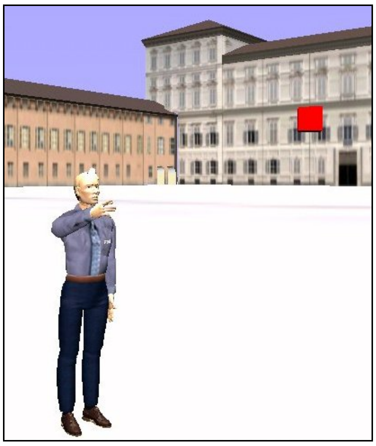


Figure 27 Example of AML Pointing

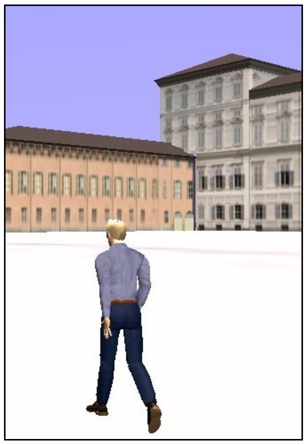


Figure 28 Example of AML Walking

3.3.4.4 TTS Engine

The used TTS engines, which have been provided by Telecom Italia and France Telecom, fulfil our requirements. They produce adequate audio streams and phoneme sequences with timing information. They are also able to output a MPEG-4 fap file or a viseme file (representing lips movements), in addition to, or as an alternative to, the phoneme sequence. Furthermore, it is possible to annotate the input text with emotional tags.

3.3.4.5 FBA and Audio Encoders

FAPs and BAPs extracted by the AML Processor from the AFML and ABML parsers, are then passed to the Face and Body Animation (FBA) encoder. The output of this module is binary FBA data that merges the animations generated independently by the aforementioned modules. As previously described, the FBA output is subdivided into Access Units (AU), each associated with a BIFS-anim compliant header. Similarly to the FBA stream, the encoded audio can be transmitted over a MPEG-4 compliant network, using a dedicated streaming server.

In the current implementation, the FAPs and BAPs are finally encoded and transmitted across the network from server to client. However, it is possible for AML itself to be used for communication and for the AML processor to be implemented on each client. Although AML was not designed for this particular purpose, this will reduce the bandwidth requirements. However, this puts two constraints on the client: availability of the FAP DB and the BAP DB, and implementation of the AML processor on the client. Also, AML needs to be standardized to be used in this manner.

3.3.5 Conclusion

In this section, we have proposed a new animation language describing complete avatar animation, consisting of face and body animation and Text To Speech as the basic elements. The choice of a XML-based syntax greatly facilitates the validation (e.g. the ABML Schema is provided), parsing and streaming of the scripts.

The important features of AML are:

- It is a high-level language to be used by intelligent agents driving 3D avatar geometries. The scripts can be predefined or generated on-the-fly by the agents.

- It is easy to use by designers for the creation of animations. It is flexible enough to create a rich variety of animations, even though it uses basic predefined animation blocks.

- It allows the generation of seamless animation using facial expressions, gestures and TTS, and also gives explicit control over their mutual synchronization.

- AML also has a potential to be used to stream avatar animations over low bandwidth networks, where conversion to low-level parameters and encoding are not required.

Owing to these features, we think that AML is a valuable tool for the simulation of social interaction in virtual environments.

3.4 Example Application: A Virtual Assistant Interacting with Real Users

The various components described in this chapter have been successfully integrated in the SoNG platform (*SoNG homepage*). SoNG's objective was to investigate, develop and standardize the building blocks for the next generation of portals. Portal is a term synonymous with an access point to resources and services on the Web. Typical services offered by portals include directory of resources, search facilities, news, e-mail, voice chat, map information, and sometimes community forums. For the time being, these services generally rely on point-and-click on structured information like text, still pictures or 2D graphics.

According to SoNG's philosophy, the new generation of portals should allow end-users to access resources and services in an easy and natural way, by enriching the classic 2D web content with 3D computer graphics elements like embodiments and virtual environments, intelligent agents, new user-interfaces for rich human-agent interaction and real-time audio-visual communication. For this purpose, SoNG's technology integration platform is a fully MPEG-4 compliant player, used as a web browser plug-in, which is based on the platform provided by Blaxxun Interactive (*Blaxxun Interactive homepage*; *Octaga homepage*). It was developed further in collaboration to incorporate all the functionalities required.

In order to demonstrate how these new technologies can be integrated, a sample e-commerce application was developed in the framework of a dedicated SoNG work package headed by the Imperial College. Letting aside the audio-visual aspects implemented by other partners, we present our scenario of an intelligent virtual assistant guiding users in a rich shopping metaphor. A human user represented by a 3D avatar can navigate in a scene representing a telephone shop, and is accompanied by an embodied autonomous agent which is able to answer his/her textual queries and, for example, to point at a product while describing verbally its characteristics. This application illustrates the integration of MPEG-4 and AML components on a PC platform, but also constitutes our first experience of the generation of embodied social interaction using autonomous agents.

In this section, we discuss some properties of human-agent interactions, present the results of our requirement analysis, and give an overview of the resulting agent architecture and of its integration in the SoNG platform.

3.4.1 Human - Agent Interaction

As it appeared in the experiment described in the preceding chapter, humans' interactions with one another are multimodal, i.e. composed of speech, facial expressions, gestures, postures, etc., and are governed by a set of specific social rules. Communication between human and current computer systems using classic interfaces obviously does not work in the same way. According to Reeves and Nass (1996), humans have evolved to communicate with one another and are very efficient in adapting their interaction to context and in filtering out information. Standard direct manipulation interfaces can be successfully used because of great adaptation abilities. However, all individuals are not equally competent: many people who are able to interact normally with other humans have difficulties manipulating computer interfaces. Moreover, a learning period is still required for all individuals. By making computer interfaces more human-like using embodied intelligent agents, it is hoped that the communication overhead is reduced and that the computer systems become more engaging. Consequently, research in animated interface agents has gained a lot of interest in recent years.

However, this idea has also been strongly challenged by a number of critics (Norman 1997; Maes & Schneiderman 1997; Dehn & van Mulken 2000). Three main arguments are given:

- Control should lie entirely with the user or it may lead to anxiety.
- Anthropomorphising can result in false expectations of the actual capabilities of the system.
- Animated graphics can be more of a distraction from rather than an enhancement of interaction.

Although these points are valuable and must certainly be taken into account in the design of human-computer interaction systems, we don't think they have a sufficiently far-reaching impact for the project to be dropped altogether. Moreover, a number of studies and usability evaluations (van Mulken, André & Müller 1998; Koda & Maes 1996; Moon & Naas 1996; King & Ohya 1996; Lester et al. 1997a; Charlton & Kamyab 2000) underline the advantages of using animated interface agents. An evaluation of DFKI's PPP Persona (André, Müller & Rist 1996), by van Mulken, André and Müller (1998) demonstrates that the use of animated interface agents has no detrimental effect on the objective performance of the user and that the experience is perceived as less difficult and more entertaining. Koda and Maes (1996) observe that systems which use animated faces in their interface are rated more likable and enjoyable. Lester et al. (1997a) show that an increase in students engagingness and motivation, leading to improved performance, occurs when animated agents are used in pedagogical learning environments. This important phenomenon is referred to as the "persona effect".

In order to produce natural face-to-face interactions of human users and embodied intelligent agents, a number of important features must be provided in the system. Here are the required features we have identified, some of them logically similar to the ones described in the preceding chapter for human / human interactions in virtual environments:

- realistic human embodiment for the agent,
- simultaneous usage of multiple communication channels (voice, face, body),
- reactive and proactive communication acts,
- a number of social rules must be respected

MPEG-4 and AML can be used to help fulfil the first two requirements. However, in order to allow easy-going cooperative dialogues, additional natural language communication facilities should be provided on the user side. In SoNG, we allow users to type natural language queries or commands in a chat window. To reply, the agent "speaks" through a TTS engine, making the interaction similar to a conversation with another human, but also uses the same chat window in order not to force the user to switch communication channel. In addition to this feature, it is important that the agent be able to keep track of the user's conversational state and of the interaction context. It would also be valuable that the agent pick up clues about the user's emotional state from their conversation.

In contrast with standard computer interfaces which are purely reactive, the intelligent agent must be able to produce proactive behaviours. It should not only answer queries, e.g. give the price of an item, and execute orders, e.g. purchase a telephone on the user's behalf, but also attempt to interact with the user when it deems it can make a valid contribution. For example, it could suggest an item which matches particularly well the user's profile. This personalization of services requires that the agent manage a database of user preferences. Other examples of proactive communication could be to simply ask the user how he is doing after a long period without interaction, or to gather information about his/her identity, preferences, etc. Reactive behaviours may also be triggered by events and environmental changes as well as user actions which are not directed to the agent but require its intervention.

The social rules to integrate in the agent's reasoning apparatus are primarily related to the structure of the interaction (as described in the preceding chapter, signalling the beginning and termination of an interaction, turn-taking, etc.), but also to the nonverbal content itself. For example, men commonly use gestures that are different to greet other men (e.g. hand-shakes) than to greet women (e.g. kiss on the cheeks). If this kind of rules are not taken into account in the design of the agent behaviours and inadequate behaviours are triggered during the interactions, it is probable that the user's comfort level will decrease significantly. Even more fundamental is the fact that, depending on the social and cultural context, nonverbal messages are not understood in the same way, which may lead to misunderstandings (e.g. the "thumb up" gesture means "good job" to many people but is taken as an insult in some parts of the world). This means that social characterization of the users, embodied agents, and behaviours, is required. Figure 29 gives an overview of a behavioural architecture fulfilling the described requirements.

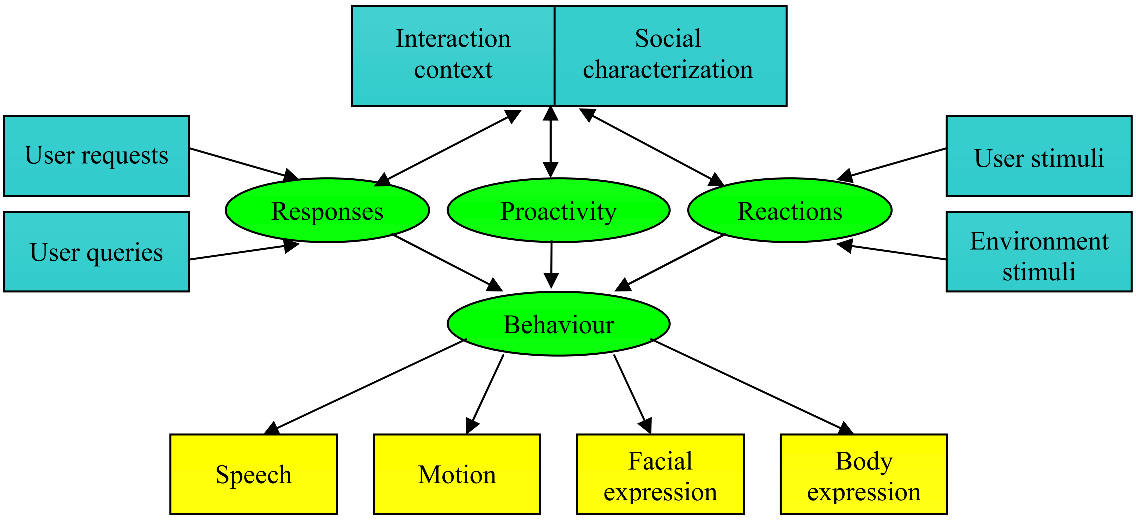


Figure 29 High-level behavioural architecture for human-agent interaction

3.4.2 High-level Behaviours Specification

SoNG agents used in the e-commerce application must be able to take on the role of personal assistant or sales assistant. The personal assistant interacts in a friendly way with the user, shows empathy, helps him/her when difficulties arise, acts as a delegate, etc. Generally, its task is to lower the "getting started barrier" and to increase the user's motivation by making his/her experience more enjoyable. In the case of the virtual sales assistant, which mainly manages the commercial transactions using the "contract net" protocol, a graceful handling of the interaction with the customer allows the building of trust and of a certain degree of intimacy, which are very important for commercial activities. The customer is recognised and engaged in a friendly way, can obtain specific assistance or can be directed throughout the whole buying decision tree.

The shopping metaphor requires the replication of everyday behaviours that are important in any social interaction, e.g. nonverbal greetings, but also the implementation of specific behaviours used to facilitate commercial transactions or guide the user in the environment. Basing our work on use cases and requirements analysis, we organized the different behaviours to provide in a number of interactional roles to be embodied by the assistants, and described in details each behaviour, constituted of speech, face and body actions. An overview of the roles and behaviours is given in Figure 30. Our specification includes the conditions necessary for the behaviours to be triggered and the hierarchical links between the roles.

Roles hierarchy	**Attached behaviours**
General interlocutor role	Greeting, Speaking, Listening, Sleeping, Puzzled, Walking, Leaving
Sales assistant role	
Informant role	Product exposing, Describing, Suggesting, Calling
Transaction handler role	Acknowledging, Bargaining, Transaction status signalling
Personal assistant role	
Informant role	Informing, Evaluating, Curiosity, Attention directing
Emotional interlocutor role	Emotion displaying, Listening, Feedback encouraging
Navigation aid role	Following, Location informing, Guiding, Pointing
Delegate role	Mission accepting, Success signalling, Contract net

Figure 30 Overview of SoNG assistant's roles and behaviours

3.4.2.1 Social Characterization

As explained in the preceding section, social characterization is important in order for SoNG's assistants to be perceived as believable interlocutors. Sales assistants' social profile is chosen at design time. For personal assistants, the best option is to allow the user who is entering in the virtual environment for the first time, to choose among various social characteristics for his/her assistant. The profile influences the assistant's representation and communication styles, but doesn't affect its fundamental behaviour and the tasks it is in charge of. All agents adapt their nonverbal behaviour to the users' profile when it is known (generally after that the "Introductory behaviour" has been performed). Table 4 gives an overview of the selected criteria.

We have first chosen three classic social-statistical variables:

- "Culture" refers to the set of norms, rules of behaviour and body language that are specific to (or originate from) the people living in a given geographic/cultural area.

Since many different choices could be proposed for this variable, a good solution is to adapt the options to the participants' origins as SoNG's user base grows. For the first stage of the project, we chose two categories that are different enough to evaluate the approach: "Europe (Western)" and "Asia (Far East)"[3]. Gestures involving physical contact should for example be avoided by an "Asia (Far East)" assistant since it is usually negatively connoted in this area. This criterion is also used to customize the assistants' representation.

- The "Gender" criterion has two possible values, "Male" and "Female", which are used to determine both the assistants' communicative behaviour and representation. Several studies have shown that gender is an extremely powerful cue, even in the context of man-machine interaction, e.g. male synthetic voices are generally rated as warmer than female synthetic voices, but also more dominant and forceful (Nass, Steuer & Tauber 1994). Typically, the performed greeting gestures won't be the same depending on the gender choice.

- "Age" is another important criterion. We use a typology of three predefined options in order to restrict the number of possible representations and behaviours: "Child", "Adult", "Elder". This criterion can prove useful in making the interaction with the assistants more enjoyable (e.g. children users may prefer to be guided in the environment by an elder or by another child depending on their personality).

Criteria / Influence on…	**Representation**	**Communication style**
Culture	Yes	Yes
Gender	Yes	Yes
Age	Yes	Yes
Extrovert/Introvert	No	Increases/decreases the frequency of proactive behaviours
Empathy	No	Increases/decreases the frequency of "emotional interlocutor" behaviours

Table 4 Social characterization criteria

These three criteria allow the designers/user to choose between twelve different assistant models. These same criteria can also be used by the assistants to help them make appropriate suggestions. When users' and assistants' "Culture" settings are different, a possibility is that the assistants use the user's "Culture" setting in order to perform understandable gestures, but keep their own "Gender" and "Age", which should not produce misunderstandings since they are mainly used for personification.

In addition, the use of two social-psychological variables is proposed:

- The "Extrovert/Introvert" criterion is used as a metaphor allowing the designers/user to specify the frequency of the assistants' proactive behaviours, e.g. make suggestions about where to go or inform about new available products.

[3] This is of course a very simplified view since these cultures are themselves made up of different entities.

- The "Empathy" tuning is similar but specific to the behaviours described in the "emotional interlocutor" role for personal assistants. For instance, it allows the user to prevent the personal assistant from asking about his/her state-of-mind or describing its own.

3.4.3 Agent Architecture

An overview of the agent architecture is given in this section but a more complete description is to be found in Kamyab et al. (2001), Guerin et al. (2001), Kamyab (2001)[4].

SoNG agent architecture is based on JADE (Java Agent DEvelopment Framework) (*JADE homepage*), a software framework facilitating the development of multi-agent systems through a number of services related to the FIPA (*FIPA 97 homepage*) ACL, like message transport, queuing, parsing, etc. JESS (*JESS homepage*) has been chosen to implement the protocols that define the agent behaviour. It is a declarative rule engine that allows for the easy development of expert systems in Java environments, and provides a number of powerful services like forward and backward chaining, the ability to directly reason about Java objects, etc. A MySQL database (*MySQL homepage*) server with JDBC support is used to manage the agent's beliefs and information related to the application ontologies.

Figure 31 gives an overview of the agent architecture. The following components are included:

- **Natural Language Parser**: Translates user textual inputs into ACL messages.

- **Reflex Response Planner**: Generates immediate action to events, such as facing the user's avatar when a text is sent.

- **Action Planner**: Plans actions and conversations by generating AML scripts as well as new ACL messages (also generates proactive behaviours).

- **World Interpreter**: From received speech acts and exogenous events, updates the agent's beliefs.

- **Social State**: Stores social characterization data as well as other dynamic variables, e.g. the current emotional state.

- **World Model**, **User Model** and **Discourse Model**: Stores beliefs about the world, about user's recent actions and preferences, and about the current state of the conversation.

The main objective of this agent architecture is to allow personal and sales assistants to maintain a coherent conversation with a human user. In order to constrain the agent's choices at any point in the conversation, what has been done is to define protocols for each interaction, which ensures that the social conventions specific to the type of interaction are respected. The protocols are implemented with JESS rules whose syntax can be found in Friedman-Hill (2003). When a new ACL message is received, the World Interpreter checks

[4] Several important components, for example the natural language processor, are purely the work of the Imperial College.

the conditional statement of the protocol's rules. If a rule is fired, an action can be executed through the Action Planner and a change in the state of the agent's beliefs can be made. Other rules can then be triggered by this change as inference occurs.

In addition, a specific role is linked to each rule in a protocol. Two types of roles can be provided: social roles and interaction roles. The social role is permanent. It is primarily related to the general function of the agent, e.g. Shopkeeper, but social characterization can also be included, e.g. Male/Adult/European/Shopkeeper. On the contrary, an interaction role can be assigned at any moment to an agent when a specialized behaviour is required. The roles described in the preceding section are implemented as interaction roles, e.g. the "Delegate" role. A rule is triggered only when the agent's role matches the role specified in the rules' conditional statement.

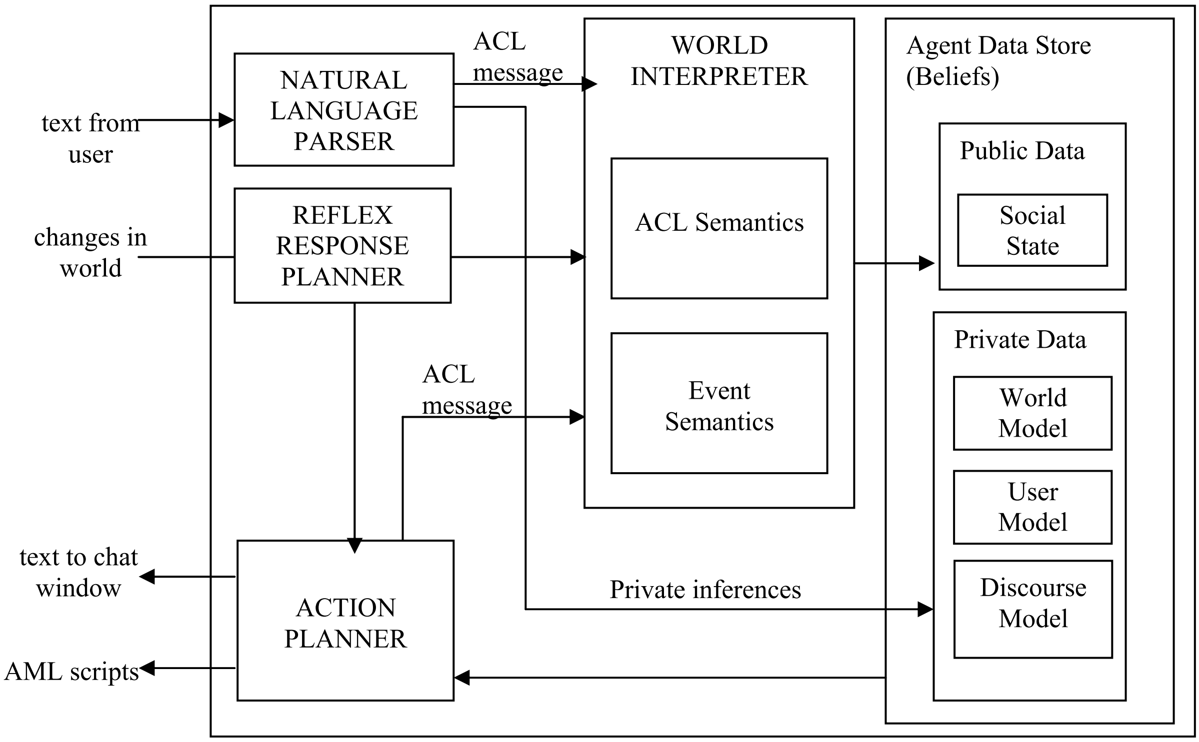


Figure 31 Overview of SoNG's agent architecture

Since the FIPA semantics was not well suited to the application requirements, we designed our own message classes embedded in FIPA inform messages. The following items are included in the syntax: sender, receiver, performative, content, conversation_id, sequence_number. "Sender" and "receiver" identify the involved agent / avatar, "content" is expressed in the agent's simple internal language, "conversation_id" identifies the current conversation and "sequence_number" defines the current state of the interaction. The available performatives are: accept, inform, propose, query, reject, reply, request. Figure 32 shows the syntax of our ACL messages.

An example of action expression is given in Figure 33. It corresponds to the desire of an agent engaged in a shopkeeper-customer interaction to communicate a suitable list of items to the user. When the code is parsed, “role:customer” is substitued with an internal ID and the date of the last interaction can be computed (“(find list_of: telephone: date:2000-02-27 01:52:27)”). Then, a search agent is in charge of querying the database. The second action, which also requires a preliminary substitution phase (“(tell John list_of: telephone: price:preference(John)”), further filters the list of items and triggers the required behaviour.

```
"myphoneagent@frank1:1099/JADE","mysearchagent@frank1:1099/JADE",query,
"list_of: telephone: date:2000-02-27 01:52:27 ","myphoneagent@frank1:1099/JADE2",1
```

Figure 32 Example ACL message

```
"(in shopkeeper-customer
                    (find
                      list_of: telephone:
                        date:last_interaction(in role:customer)
                    )
                    ;
                    (tell
                      in role:customer list_of: telephone:
                        price:preference(in role:customer)
                    )
)"
```

Figure 33 An example of action expression

Figure 34 illustrates the functioning of the system when the avatar of a human customer enters the shop model in order to buy a product.

If an agent occupying the social role of shopkeeper is present in the shop at that time and if no interaction already existed for the two entities, a new interaction state is created (step 2). Immediately, a rule is fired that triggers an AML greeting gesture, which is mandatory at this stage. The shopkeeper’s obligation to greet is discharged and the interaction state is incremented. Step 3 is a branching point where several options are available. The user may have already asked for a specific item, in which case his/her query must be handled in priority. Otherwise, the assistant may decide to propose a special offer. In both cases, the interaction reaches Step 4 after some iterations and the process continues until the end of the protocol is reached or an interruption takes place.

In order for the Action Planner to be able to easily create suitable AML scripts, a database of predefined AML models is available. The models are classified according to the protocol name, agent role, and interaction state. Before sending the script to the AML parser, the Action Planner loads the adequate model and replaces some dedicated tags with relevant information. Typically, if the script to trigger contains a Facing action, the current coordinates of the user’s avatar are inserted.

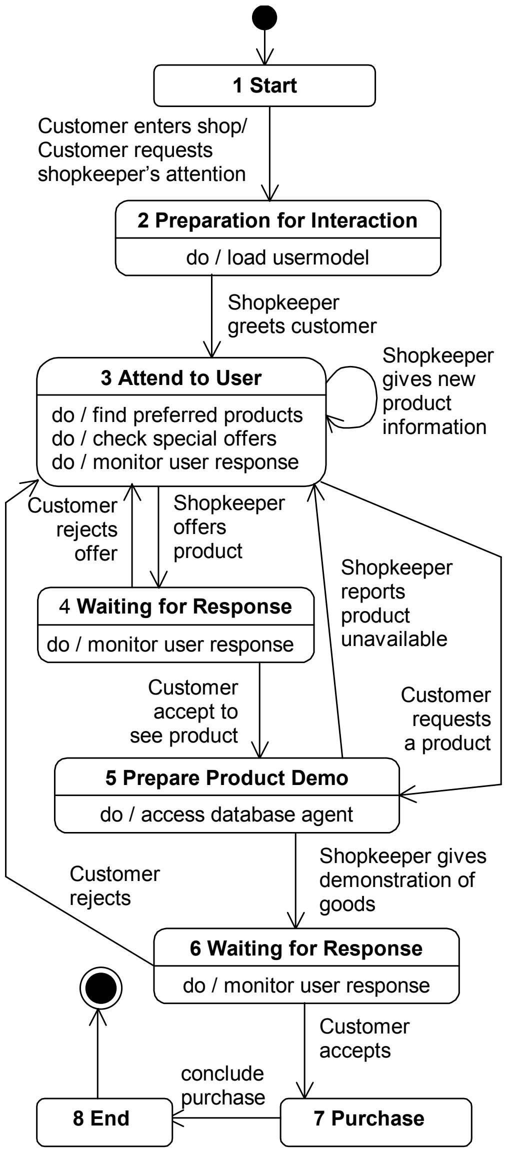


Figure 34 Example interaction between a virtual sales assistant and a human customer

3.4.4 Integration

Figure 35 shows how the different components we have described in this chapter are integrated.

In addition to the client side agent we have just described, a server side agent service has been added. It allows the update of several SoNG clients, typically the clients connected to the same scene, through a MPEG-4 compliant streaming server. The service receives high-level commands including AML content and calls the processor. The AML processor is here situated on the server side but an alternative scenario where AML scripts instead of FBA data are transported, would also be applicable. When the client receives the encoded MPEG-4 stream containing FBA and audio data, the corresponding decoders are activated. The FBA decoder extracts BAPs from the FBA stream and sends them to MPEG4SkeletonLib which updates the scene graph.

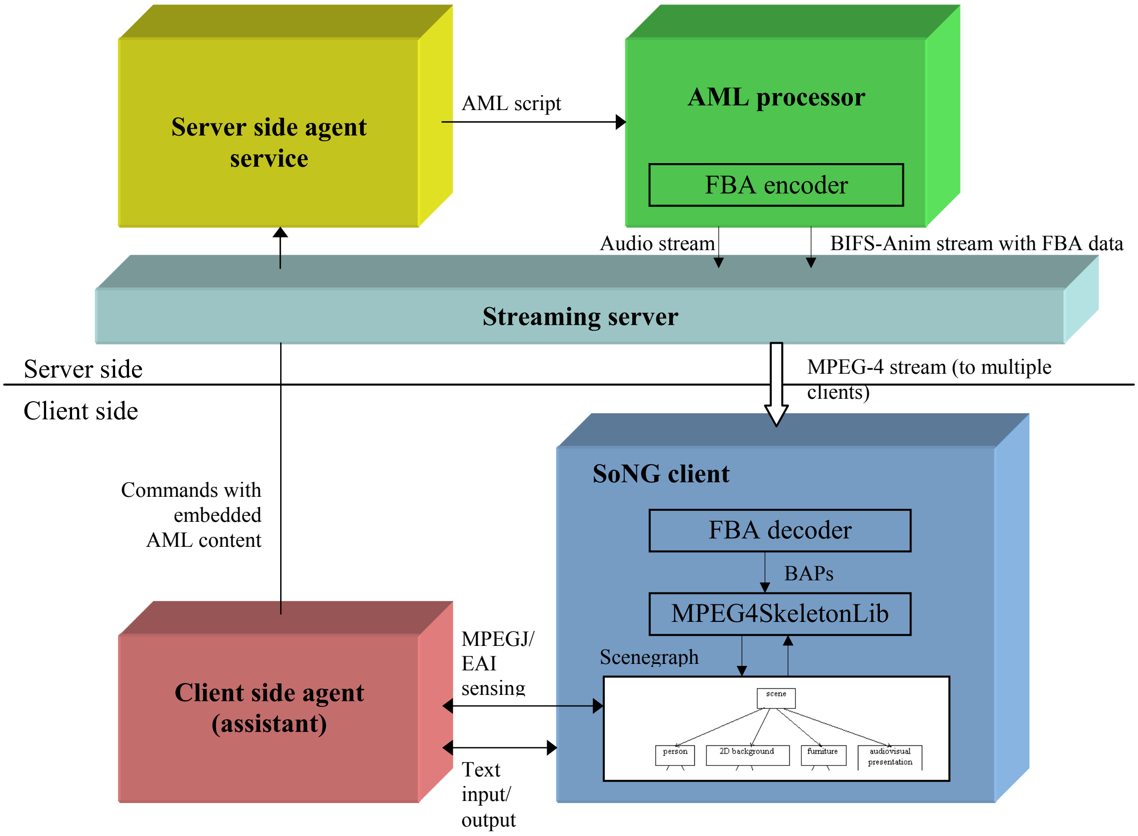


Figure 35 Integration of MPEG-4 components, AML and agent technology in SoNG

Since the agent has to react to certain events that occur in the virtual world, it must be able to sense the 3D environment by scanning the scene graph. This can be done using the MPEGJ/EAI programmatic interface. Such events as the arrival of a new user or the clicking of an object can be detected and notified to the server side agent. Another possibility is to enrich the scene definition itself with sensors, like touch sensors or proximity sensors. Server commands are then generated under precise conditions, and the agent can react appropriately. The chat window being a simple Java applet, the capture of the user's textual inputs and the sending of the replies are straightforward.

The FBA encoder/decoder and body animation components are integrated as dynamic libraries (DLLs). MPEG4SkeletonLib connects to SoNG client's scene graph using the Adapter model previously described.

3.4.5 Example Application

Figure 36 contains several snapshots of a 3D e-commerce application which illustrates the integration of AML and of the described agent technology with the SoNG player. When a user's presence is detected in the 3D shop, the agent-controlled sales assistant greets him before presenting verbally and nonverbally a cellular phone model. After the user has informed the sales assistant that he is not interested in the purchase, they move to another product which is also described. Finally, the virtual sales assistants leaves, walking sadly, because it has not been able to make a deal.

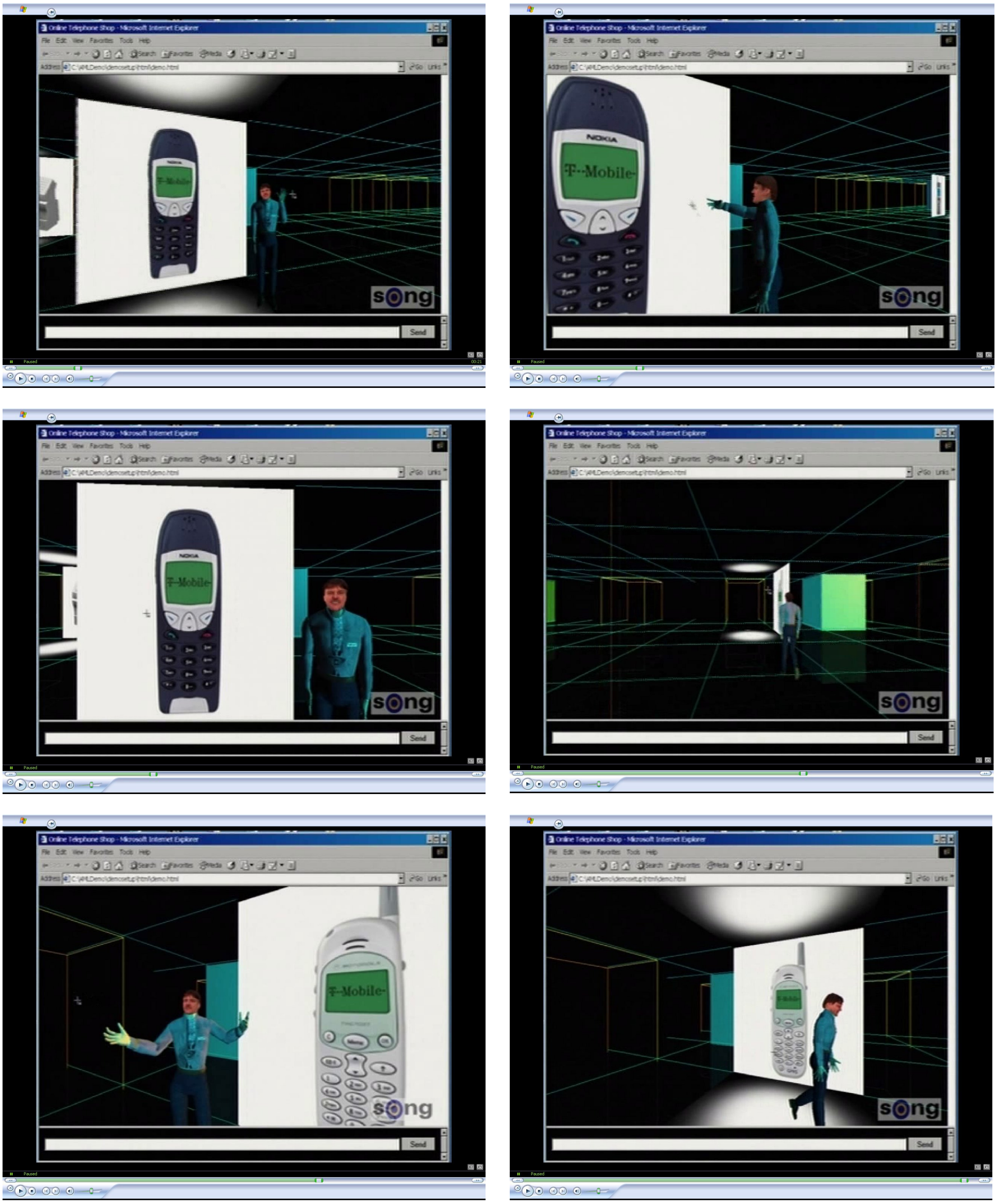


Figure 36 An agent-controlled sales assistant trying to sell cellular phones to a human user in SoNG

3.5 Conclusion

In this chapter, we have shown how standardized MPEG-4 technology can be used to visualise embodied social interactions. We also explained how the low-level MPEG-4 services can be enriched in order to facilitate the fine specification of participant’s nonverbal behaviours, and introduced, for this purpose, the Avatar Markup Language. We also gave an example of the integration of these components with agent technology. We feel that the work presented fulfils the low and high-level requirements discussed in the chapter's introduction.

Our experience with MPEG-4 in the SoNG framework was very enriching. In accordance with other critic reviews (Preda & Preteux 2002), we argue that it is an appropriate platform for FBA animation, based on sound and efficient techniques, and which has all the advantages of a standardized technology. However, MPEG-4 is also a very large and complex standard, and the development of a fully compliant player requires a significant effort in conception and programming. In particular, the streaming and encoding/decoding of different compressed stream formats (FBA, BIFS-Anim, BIFS-Update, MP4, etc.), sometimes embedded in one another, can create some development overhead. This may be related to the hybrid nature of the MPEG-4 standard, which deals at the same time with natural pixel-based images (video) and synthetic computer-generated objects (3D models), with a strong emphasis put on content delivery.

The development of SoNG's virtual assistants was an interesting opportunity to experience the creation of embodied social interactions using autonomous agents. AML allowed the introduction of rich face and body animations, however, the agent's ability to correctly handle a (textual) conversation with a human user, which was required in priority given the application context, greatly influenced the design of the agent architecture. For example, the translation of natural language in ACL messages that can be interpreted by the agent and the production of textual content corresponding to ACL messages, one of the main problems in human-agent interactions, was a central task in the project. Our next developments, which took place in a different context, focus more specifically on the handling of nonverbal behaviours.

In order to improve the quality of the services provided to the users, we enriched our agents with some simple social characteristics and introduced in the behaviour specifications a first concept coming from the social sciences, the "social role". We are now going to further develop our ideas of the social modelling of embodied agents. In contrast with the application just presented, the emphasis in the following chapters will be on nonverbal agent-agent interactions rather than human-agent interactions, and on simulation rather than task-oriented behaviours.

Chapter 4 ● Modelling of the Social Agent: An Enriched Agent Architecture based on the *Homo Sociologicus* Model

4.1 Introduction

It appeared in the preceding chapter that agent oriented software design is well suited to the control of complex interactions. The use of autonomous agents to handle social interactions seems all the more natural as the name "agent", or "social agent", has been used for a long time in sociology to characterize the member of a group or society. In this chapter, we discuss the suitability of agent oriented approaches for the simulation of human social interactions and social behaviour in general, and propose an enriched agent architecture dedicated to this purpose.

Most existing agent architectures are either dedicated to the fulfilment of specific tasks requiring planning and other problem solving techniques, as in the works of Ingrand, Georgeff and Rao (1992), or to the simulation of emotional behaviours (Botelho & Coelho 1998). In contrast with the state of the art, pure goal oriented and emotional behaviours only constitute, according to social scientists, a small part of our daily activities as human beings. Everyday, we engage in many interactions and social activities, dealing with contradictory motivations, adapting our behaviour to the cultural context, and also simply following routines. Occasions where human behaviour is primarily related to an emotional state or is the result of an explicit problem-solving process, are comparatively rare. As will be discussed later, in a sociological perspective, individuals mainly act in order to meet other's expectations, according to a learned definition of what is a normal or appropriate behaviour in a given situation. Group belonging, gender or cultural origin, for example, are important determinants in this process.

These facts need to be taken into account in the development of virtual humans as socially intelligent virtual agents, capable of adapting to and interacting realistically with other agents in 3D environments. We think that the next logical step in the development of such agents is to model their behaviour at a higher level, adding rich social identities and social motivations on top of their specific competency.

Such expressions as "socially intelligent" and "socially adept", have begun to appear in the agent literature (Dautenhahn 1998) . Two other important characteristics of social agents can be added:

- They should be “socially motivated", i.e. acting according to their social identity and position in a group/society.

- They should be “socially believable", i.e. ideally, an external human observer, using his/her own cognitive resources, should perceive the established connection between the agent's identity, the social context and the agent's behaviour, as meaningful.

In our view, the classic opposition in AI approaches between thinking/acting humanly and thinking/acting rationally is replaced by the goal of allowing agents to think/act in a "humanly rational" way. This means that a number of mechanisms must be provided to allow socially situated agents to process the interaction context using their own specific world views, extracting relevant information in order to adopt a behaviour which is "correct" in terms of social values and social acceptance.

In the next sections, we try to identify, using sociological literature, some of the most important components and mechanisms that should be included in an agent implementation dedicated to the simulation of human social behaviour. We present a conceptual architecture based on the so-called *Homo sociologicus* model, whose components constrain the action selection process of a socially situated agent. It has been inspired by the BDI approach (discussed in the next section), but allows the simulation of a number of social phenomena, like norm-based behaviours. Interestingly, the work simultaneously conducted by Dignum et al. (2000) and Panzarasa, Norman, & Jennings (1999) shows very similar objectives. The high-level architecture presented is not a full agent architecture *per se*, but it can be implemented as an extra layer driving the agent's standard reasoning apparatus.

This chapter is organized as follows. In Section 4.2, we compare two logics of human action which are dominant in the fields of computer science and sociology, and discuss their implications for the design of social agents. Section 4.3 presents a number of useful concepts constitutive of the *Homo sociologicus* model as well as a list of functional requirements. In Section 4.4 we give the specification of the architecture. Section 4.5 describes a prototype implementation and provides a simple example of produced behaviour.

4.2 Models of Action for the Modelling of Autonomous Agents

As noted by Dautenhahn (1999), the conception of sociality varies in different academic communities, and this has a direct impact on the modelling of social agents. While Dautenhahn tackles the issue from a psychology and ethology viewpoint, we use the sociological tradition to question the mainstream approach of agent oriented design in computer science.

4.2.1 Homo Sociologicus vs. Homo Economicus

We have already mentioned that the term "agent" is commonly used in both computer science and sociology. However, this apparent connection does not match very well the actual use of the concept in the two fields. In computer science, this term enforces the idea of autonomy and proactivity (Wooldridge & Jennings 1995). In sociology, on the contrary, the word "agent" is often used in opposition to "actor", in order to underline the fact that human beings are social beings who cannot be properly understood without referring to the societal and cultural context which constantly shapes their behaviours (Bourdieu 1990).

"Objects do it for free; agents do it for money" (Jennings, Sycara & Wooldrigde 1998). This mantra, which can be frequently heard in the agent community, is another clue which points to an important difference in the dominant paradigm of these fields: most agent oriented approaches are based, explicitly or implicitly, on a self-interested conception of human action which is often referred to as "*Homo economicus*", while sociologists have, along the years, developed a series of alternative arguments and concepts that can be grouped in a model called "*Homo sociologicus*".

Here is a possible definition of the corresponding ideal-types:

- *Homo economicus* is a self-interested actor which consciously pursues a number of precise goals. It uses rational calculation of costs and benefits to evaluate all possible behavioural options, and picks the optimal solution in order to maximise its profits and satisfy its needs. Its choices are unconstrained by social norms and obligations.

- *Homo sociologicus* is immersed in a social environment which determines its behaviour. It doesn't act to pursue selfish interests but to fulfil social roles which it has acquired through a socialization process. Confronted to several options, he chooses the most appropriate course of behaviour given its perception of the situation and the social roles it has internalised.

More detailed explanations can for example be found in contributions by Archer (2000), Ley and Johnson (1990), Kirchgässner (1991). These two logics of action, which affect approaches like game theory (Axelrod 1981) and Bayes decision theory (Berger 1985), have been, and still are, intensively discussed in several fields. We do not have the ambition to resolve this complex dispute, especially as Kuhn (1962) and Feyerabend (1962) have underlined the far-reaching implications of the two models. We are mainly interested in the influence that the *Homo economicus* model has, and the influence that the *Homo sociologicus* model could have, on the way autonomous agents are designed.

4.2.2 BDI Architecture and Social Simulation

The belief-desire-intention (BDI) model of rational agency (Bratman 1987; Wooldridge & Jennings 1995), because of its explicit use of mentalistic notions, is well suited for a comparison of both approaches. The central position of the concept of "desire" in the BDI architecture, i.e. it is the unique source of agent motivation, makes BDI agents strongly related to the self-interested *Homo economicus*. Systems built on the BDI architecture have produced interesting results in complex environments e.g. dMARS (Ingrand 1992). It has proved to be a useful abstraction to model autonomous agents and offers a convenient and intuitive way to structure the agent's action selection.

However, we agree with Balzer (2000) when he points out that BDI architectures are deficient for the simulation of social behaviour. Like every system based on the *Homo economicus* model and related to the Bayesian decision theory, they do not take into account the socially situated nature of human behaviour. More specifically, a number of important features are missing in BDI systems:

1. social constraints and power,
2. moral standards and benevolent action ,

3. norms and routine behaviours

It results that a number of extremely common situations cannot be properly simulated, for example:

1. A social agent may behave in a way that is contrary to its desires because of the external influence of one or several other agents, generally endowed with a higher social status.
2. A social agent may not behave in a way that matches its desires, because it perceives such a behaviour as morally "wrong", or because it has to provide assistance to another agent because it is morally "right".
3. A social agent may adopt a behaviour without considering whether it corresponds to its desires or not.

This last point is particularly fundamental, at least conceptually. Every action taking place in the main loop of a BDI system follows a decision resulting from deliberation. On the contrary, *Homo sociologicus* is not trying to make a decision: it is primarily engaged in a "process of fitting and recognition" (Balzer 2000), checking if the behaviours it has learned apply to the situation, without calculating the consequences of its actions.

The development of "benevolent agents", which "adopt other agents' goals and interests" (Conte & Castelfranchi 1995), partly addresses the other mentioned issues. In the framework of a BDI architecture, it is technically possible to declare as desires the obligation to carry out external instructions, or to adapt to some moral standards. However, we think that it results in the dissolution of the concept of "desire" itself and introduces a significant confusion in the behaviours specification. The evolution capacity of the system could be severely restricted. More generally, when a benevolent agent is designed as an entity "accepting all requests made" (Jennings & Kalenka 1998), it does not correspond to an alternative type of social motivation, rather it is simply a way of transferring the motivational apparatus to another agent. We think that when the design of benevolent agents is based on a strong theoretical background, like in the work by Mohamed and Huhns (2001) where benevolence is motivated by a combination of societal norm and long-term individual profit, it has far more reaching implications from a social simulation point-of-view.

Interestingly, the *Homo economicus* model itself doesn't fully address the issue of actor's goals. Since their origin and conditions of production are not specified or explained, it can be said that goals are exogenous rather than endogenous to the theory. In BDI applications, it is up to the designer to directly fill the system with a series of adequate "desires" allowing the agent to provide the required service. These *ad hoc* desires are generally unconstrained and unstructured, i.e. they are not related to each other, are possibly contradictory and interlocking. On the contrary, the *Homo sociologicus* model places the source of motivation in a situated socialization process and, as we will see in the next section, provides a framework to organise different motivational components.

We feel that the conceptualisation of a high level architecture for social agents could greatly help in their modelling, in the same way as the BDI architecture does for standard autonomous agents. Such an architecture would allow for the meaningful organization of all information required for the simulation of social behaviour using the *Homo sociologicus* paradigm.

Before concluding this section, we also would like to mention that, if the *Homo sociologicus* description we provide constitutes an adequate representation of the mainstream conception of individual motivation in sociology, a number of alternative approaches co-exists within the field. Methodological individualism (Watkins 1957), for example, a school which favours the intentional stance for its analysis of human behaviour, is more directly related to the *Homo economicus* model. Similarly, there are some research efforts which are based on the application of game theory to sociology, e.g. the works of Boudon (1979). However, such approaches are not deeply rooted in the sociological research tradition. It is therefore somewhat ironic that, when the social dimension of agent behaviour is tackled in an artificial intelligence context, it is frequently restricted to its game theory aspects. The so-called "prisoner's dilemma" (Axelrod 1981) is a classic example used to illustrate agent sociality. But what is fundamentally the "prisoner's dilemma" ? A problem where a self-interested individual tries to maximize its gains; *Homo economicus* again. According to Dautenhahn (1999), Axelrod himself was aware that the explanatory power of game-theory was not sufficient for the complete modelling of human behaviour.

4.3 Relevant Concepts and Requirements Analysis

In this section, we shortly present a number of sociological concepts constitutive of the *Homo sociologicus* model, and discuss their implications for the architecture's requirements.

4.3.1 Social Role

"Role" is certainly the most central concept in the *Homo sociologicus* model. It has been developed by several sociologists like Linton (1936), Parsons (1951b), Merton (1959) and Goffman (1959), to account for the regularities in face-to-face interactions and to explain the relationship between the individuals' behaviours and social status. A role is compound of a coherent set of standard behaviours, but also includes other elements e.g. a specific world view, which distinguishes it from its dramaturgic equivalent and makes it a central component in our architecture. A role is linked to such variables as age or gender, to a professional status, etc. An important characteristic is that individuals master several roles and use them successively, e.g. a woman who alternatively behaves as a secretary, as a mother and as a football fan. Cicourel (1973) has even shown that each participant adapts regularly his/her role to the other participants' active role during social face-to-face interactions.

The use of a Role component seems a good way to organize social agents' behaviours: instead of assigning behaviours directly to an agent, it is very advantageous to build these behaviours into roles, and then attribute them to the agent. We propose to use this concept of Role because it is a coherent and documented way to group behaviours, and greatly facilitates behaviours' reuse. Since the concept is widely used in field studies, it should be possible to easily integrate specific role descriptions into the system. Finally, it is a scientific concept which has the advantage of being intuitive enough to be easily used by designers who are not social scientists. An example of the use of social roles has been given in the preceding chapter, but, as we will see, the concept can be further exploited, for example to bring specific cognitive resources to social agents.

4.3.2 Social Norms

Social norms are socially and culturally situated standards for behaviour, which guide groups' and individuals' action by specifying what must be done and what is forbidden in a given situation. A norm is fulfilled because it has been integrated by the individual and to avoid sanctions. Some norms are widely shared within a culture while others are specific to a position in society, a group, etc. Organised in systems, norms constitute a mechanism of social regulation which has been studied by sociologists since Durkheim (1895).

According to Balzer (2000), social norms can be translated into simple rules: people constantly check if the situation fits some precise conditions, and if it does, apply the rule by behaving in a precise way, e.g. open the door for a woman. As previously mentioned, this phenomenon is a recognition process rather than a decision making process.

It makes sense to attach norms to roles, even if some of them are widely shared within a culture. One could for example build a “Swissman role" and an “Englishman role", containing basic behavioural norms (e.g. rules of courtesy). These roles would be assigned to the agents together with, for instance, a professional role, which contains its own norms (e.g. to dress properly, wear a tie).

4.3.3 Social Values

A Max Weber's (1968) famous typology introduces the orientation of the action to a "value", along with the rational orientation to individual ends, the affectional orientation and the traditional orientation. Parsons (1951a) also uses intensively this concept, for example as the basis for his classification of social systems.

Integrated by individuals, values constitute a personal moral system. An action selected on the basis of a value is believed to be "right" independently of its chances of success. Values involve the conscious pursuit of an ideal abstract state (e.g. "I give it back, because I am an honest person") and also work as evaluation and classifying criteria ("I don't like him, he's a crook"). The conscious or unconscious pursuit of social values can be a powerful source of motivation, which is in competition with self-interested desires in the process of action selection. Again, some values are almost universal but others are specific to a social milieu.

The value-oriented rationality has to be implemented because it is an important specificity of human beings, which is missing in all systems based solely on the principles of the Bayesian decision theory. It is completely independent from any kind of "utility calculus". Available agent behaviours may be "rated" according to individual values, which could contribute to activate or inhibit them, depending on the agent's profile. Since values are also used to evaluate external objects and events, they can also be very useful to generate the agents' specific world views. It is very convenient to attach the values to roles: for example, "discipline" could be a strong value in a "military role", and "efficiency" may be an important classifying criterion to an agent with a "manager role".

4.3.4 World View and Typification

Identifying the "world view" of groups and how it is created and maintained, is an important task of microsociology, because it helps understand the actions of the individuals. Thomas's classic observation that one does not act according to the objective situation, but on the basis of one's "definition of the situation" (Thomas 1923), demonstrates that cognitive mechanisms act on the perception to generate specific world views and potentially very different reactions to the same situation. Bourdieu's "habitus" concept describes an abstract entity which fulfils this task: it is a "system of acquired dispositions functioning on the practical level as categories of perception and assessment or as classificatory principles as well as being the organizing principles of action" (Bourdieu 1990, p. 13).

In a different approach, ethnomethodologists like Garfinkel (1967) have also emphasized the role of categories of perception, by describing the "typification" mechanism through which we organize our perception of others and of the world. The "type" helps give meaning to the world (e.g. "she's a woman, she's certainly shy") but, in a reflexive way, its definition is also constantly updated according to one's personal experience (e.g. "women may not be that shy after all").

Typification is certainly a very important cognitive mechanism for social behaviour, and should be in some way implemented. There is no need to produce a complete world model by designing a complex ontology based on the "is-of-type" relationship, only the most important evaluation criteria for each role should be provided. Beliefs about other agents are particularly crucial to generate on the basis of their roles (e.g. for an agent with a "salesman role" perceiving another agent: “Is he a client ? A colleague ? The boss ?"). Predefined beliefs about a number of perception types, as well as the way for an agent to recognise other agents' types should be provided.

4.3.5 Requirements

Here is a list of general requirements for a system dedicated to the simulation of social behaviour, which takes into account the concepts just discussed:

- **Multi-behaviour architecture**: Since people in real-life fulfil several distinct social roles and are involved in various activities, the architecture should allow several global behaviours to be assigned to one agent. These behaviours are not different ways to reach one common final goal as it is with problem-solving agents, but are self-sufficient sets of actions. They should be organized in a socially coherent way and facilitate reuse.
- **Behaviour switching mechanism**: A general mechanism that is in charge of switching between these global behaviours should be provided. The local context should be used to determine if a change in the active role is required. It includes the physical environment, the temporal dimension but also the social context, typically the roles of the agent with whom an interaction is taking place. As pointed out by Sengers (1997), behaviours transitions are important to provide when agents are interacting with a real user, to explain why they are changing their behaviour.
- **Acting according to the social environment**: When assigned one of these global behaviours, agents still have to select adequate actions to perform, possibly using planning and other standard reasoning techniques. An important requirement is that the architecture allows reasoning to take place not only according to the physical environment, but also to the social and cultural environment. Thus, agents must be able to manage specific knowledge about their social environment. In addition, the

issue for a social agent is not only about what to do, but also how to do it: e.g. should it greet this particular agent in a friendly or distant way ?

- **Basic cognitive mechanism**: As previously stated, one does not act according to the objective situation, but on the basis of one's definition of the situation. Since this has a major impact on the resulting behaviour, a minimal cognitive apparatus performing typification, possibly on the basis of the agent's values, should be provided. Agents must be able to recognise, for example using physical properties, other agent's types (e.g. female/male) in order to be able to generate adequate beliefs. As additional information is gathered about individual agents, the influence of the beliefs generated by the typification mechanism should decrease. Ideally, the types' definition itself should also be updated.
- **Standard behaviours**: In many situations, humans don't have to make decisions about what to do but act spontaneously, imitating others or repeating learned actions. This is the case with routine behaviours and norm-based behaviours which need to be included in the architecture. Social norms are especially important to provide, in the form of actions that are automatically triggered/inhibited in a given situation. In most cases, they should have a higher priority than actions linked to other types of motivation.
- **Acting according to social values**: The architecture should allow the agent's actions to be oriented to social values, and not only to the fulfilment of specific goals. This means that such criteria as "what's right", "what's wrong", "what's cool", "what's out", etc. are used to choose a behaviour independently of its usefulness. Actions should be rated according to individual values, which contribute to activate or inhibit them.

4.4 Specification

In this section we present an overview of the specification of a high-level architecture for social agents, whose conception is based on the *Homo sociologicus* model. The main prerequisite for its implementation is a rule-based production system. The reasoning aspect linked to the production system is not explained in details, since it is similar to what has been written in the preceding chapter on the use of JESS rules. Details can also be found in a paper by Caicedo and Thalmann (1999).

The resulting agent architecture should be classified as a hybrid architecture, since it brings together both reactive components (e.g. using activation energy to activate an agent's role) and rule-based reasoning (e.g. to trigger norm-based behaviours). The perceptual input is processed to activate a subset of the agent's behavioural and cognitive data; standard symbolic reasoning can then be performed on the generated beliefs to complete the action selection process. It is also hybrid in the sense that it includes a simple cognitive mechanism (typification), but is not based on the conception that a full cognitive apparatus, which would very complex to implement, is required in order to trigger adequate social behaviours.

The system has been designed in order to allow designers to easily add behavioural specifications while still favouring emergent and adaptive behaviours.

The data structures organization is based on the concept of Role. As will be specified in the next sections, a Role has the following features:

- It contains standard behaviours, values and types.
- It is part of a role hierarchy and inherits data from its parent roles.
- It can be associated to other roles.
- It is assigned to an agent with a specific weight, along with other roles.
- It is activated or inhibited by a switching mechanism.

4.4.1 Standard Behaviours, Values and Types

Roles are compound of four main data structures (Figure 37):

$$Role_i = (Name_i, StandardBehavioursList_i, ValuesList_i, TypesList_i)$$

Standard behaviours are competing plans, which specify a sequence of actions. They have the following structure:

$$StandardBehaviour_i = (Name_i, PreconditionsList_i, ActionsList_i, PostconditionsList_i)$$

Preconditions specify the conditions that must be met for the behaviour to be performed in the "correct" situation. They are predicates generally expressed in terms of the agent's categorized beliefs about the world, e.g. "*((?PerceivedAgent) has just been perceived)*", "((?*PerceivedAgent) is of type salesman)*", "*((?PerceivedAgent) honesty is low)*". Actions can involve verbal and nonverbal communication, moving to a location, manipulation of objects, etc., e.g. "*(Act (greet (?PerceivedAgent)))*". The action list can be empty when only inferences need to be generated. Postconditions induce a change in the agent's beliefs, e.g. raising of an agent's honesty rating, e.g. "*(Del (?PerceivedAgent) honesty is low)*", "*(Add (?PerceivedAgent) honesty is high)*".

Values are classifying criteria used to rate behaviours and perceived entities. A role contains a list of weighted values:

$$Value_i = (Name_i, Weight_i) \text{ where } Weight_i \in \mathbf{R}: 0.0 <= Weight_i <= 1.0$$

In addition, a value score is assigned to standard behaviours:

$$Score_{ij} = f(Value_i, StandardBehaviour_j) \text{ where } Score_{ij} \in \mathbf{R}: -1.0 <= Score_{ij} <= 1.0$$

For instance, "honesty" could be an important value for a "policeman role": it is unlikely for a policeman to break the law and he should not act in the same way towards someone he thinks is a criminal or not. Thus, a value named "honesty" is added to the role and receives a high weight - behaviours involving criminal activities will be poorly rated on this criterion - and the perceived agent's honesty score is used to write the policeman action's preconditions. It means that it is unlikely for an agent with a policeman role to select an illegal behaviour (even if the active role is another one, e.g. a "father role") and that it will manage its interactions with other agents according to their "honesty" rating. On the contrary, an agent with no role involving the "honesty" value (or with a very low weight) will behave identically with "honest" and "dishonest" agents.

Types are used to generate beliefs about the world, especially about other agents. Each role receives a list of types, and a default value score is assigned to each type:

$$DefaultScore_{ij} = f(Value_i, Type_j) \text{ where } DefaultScore_{ij} \in \mathbf{R}: -1.0 <= DefaultScore_{ij} <= 1.0$$

This means that a "woman role" may use a "man type", for example highly rated on "strength" and lowly rated on "empathy". If the system is mainly reactive, the agent will directly react to the type itself or to the type's default scores. If the agent has a synthetic memory, the man's default scores are saved at the first interaction. Then, the type is not used anymore for this perceived agent but its scores can be updated by postconditions. Depending on the implementation of the underlying reasoning system, it may be necessary to provide a mechanism translating these numeric scores into intensity categories (e.g. "very low, low, medium, high, very high").

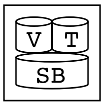


Figure 37 A role is represented with its values, types and standard behaviours

4.4.2 Roles Hierarchy

In order to allow for the incremental design of behaviours and since many roles are "specialised" versions of more basic ones, roles are organized in a hierarchy (Figure 38) corresponding to a directed acyclic graph. Child roles can have multiple parent roles, and automatically inherit their parent's characteristics, as will be described later.

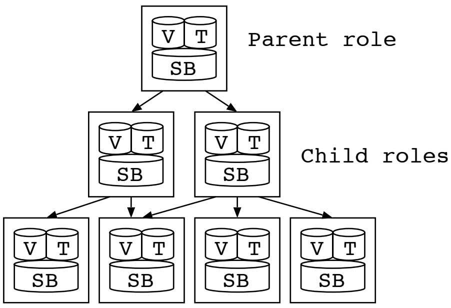


Figure 38 Hierarchical organization of roles

For instance, a "saleswoman role" can be declared as a child of a "woman role" and of a "salesperson role", but also has its own specific behaviours, values, etc.

4.4.2.1 Associated Roles

In the same way as roles can be hierarchically connected to parent roles, they can be associated to other roles, typically with a symmetrical function, as it appears in Figure 39.

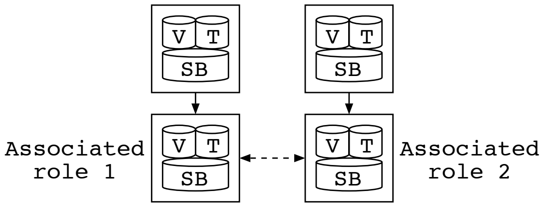


Figure 39 Associated roles

Examples of associated roles are a "mother role" inheriting from a "woman role", and a "father role" inheriting from a "man role". As previously mentioned, the use of a role can lead the interlocutor to adapt its own role: for example, a woman talking with her husband about their child, will expect of him to react "as a father". Associated roles are used to simulate this phenomenon, and to specify that a strong interactional link exists between roles.

4.4.3 Multiple Role Assignment to an Agent

Multiple roles can be assigned to an agent (Figure 40) in order to produce rich behaviours. Each assigned role has a weight that allows designers to specify its importance relatively to the agent's personality.

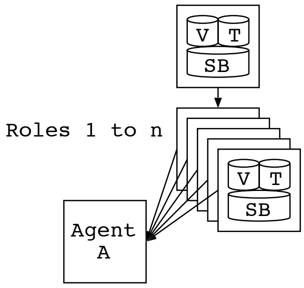


Figure 40 Multiple role assignment to an agent

When multiple roles are assigned, all their associated values' and types' scores are directly integrated in the agent's profile. Each value's weight is correlated to its corresponding role weight, and, if the same value is referenced in several of the assigned roles, a weighted average is computed. This is explained by the fact that values are intimate beliefs which are dependent on the assigned roles, but are relatively stable and don't vary when a short-term role switch is required. Perception types follow the same logic. On the contrary, behaviours are not directly integrated, but can be used when their corresponding role is active.

The values and types of the parent roles are also directly integrated, but their behaviours are searched, i.e. placed in the reasoning environment, only when the preconditions of no behaviour belonging to a more specialized role are met.

4.4.4 Role-switching Mechanism

Roles are activated and inhibited, during the execution of the simulation, by a switching mechanism which allows the agent to adapt its behaviour to the situation. If a low priority was given to the active role by simulation designers, the agent is able to easily change its behaviour when the context requires it. If the active role has a high priority, the switching mechanism does not allow a change, and the agent keeps on focusing on its current activity.

Activation energy is used to determine which role must be active. It comes from five sources:

- the initial role weight given by simulation designers,
- the location: a role can be designed to receive extra activation energy when the agent is at given locations,
- the time: a role can be designed to receive extra activation energy at certain times,
- the interactional context: a role can receive extra activation energy when its interlocutors' active role is an associated role of one of its own,
- random influence

The following operations are executed every time step:

1. The sum *A* of the activation energies corresponding to the assigned weight *W*, location *L*, time *T*, interactional context *I* and random influence *R,* is calculated for every role index *r*. It is standardised by dividing the role's energy by the total of all roles' energies, so that the total amount of added energy be identical every timeframe:

$$A_r = \frac{W_r + L_r + T_r + I_r + R_r}{\sum_{r'=1}^{n} W_{r'} + L_{r'} + T_{r'} + I_{r'} + R_{r'}}$$

   The amount of activation energy of each role is added to a corresponding summation variable *S*:

$$S_r = S_r + A_r$$

2. A decay function is executed in order to avoid the accumulation of very high quantities of activation energy that would inhibit future role switching and to enforce the provided role-based personality. Given a parameter used to determine the speed of default profile restoration $\sigma \in \mathbf{R}_+$:

$$S_r = S_r - W_r - \frac{(S_r - W_r)}{\sigma}$$

3. The role with the highest level of activation energy becomes active

The algorithm works so that the agent slowly recovers its default role priorities after having been able to respond to specific situations. The location and time criteria are useful to simulate roles linked to such scenarios as an occupation at an office or a shop, people getting out at night, etc. This switching mechanism makes possible the following scene: a man in a shop, after behaving as a classic client, begins to flirt with the saleswoman (a behaviour specified in his activated "man role"). She is able to temporarily switch from her sale role and reacts using her "woman role" behaviours, before returning to her "saleswoman role" with other clients (a change triggered by new "client role" interactions).

4.4.5 Behavioural Process

In order for an autonomous agent situated in a virtual environment to fulfil the described requirements, it must follow a full behavioural process involving perception, cognition, reasoning and action, as represented in Figure 41.

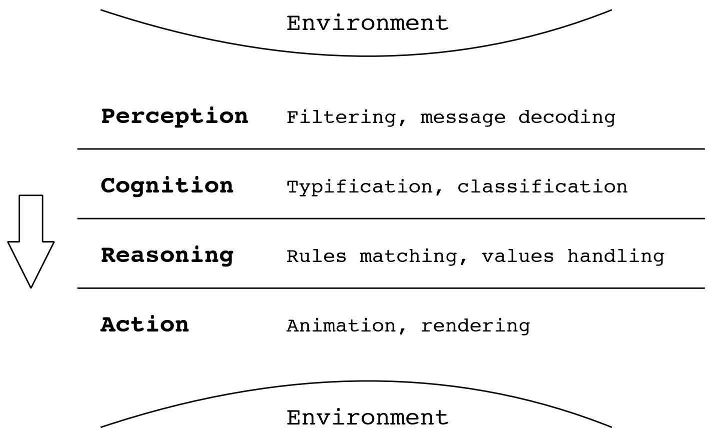


Figure 41 Behavioural process

Here are the logical modules involved in the process:

- **Perception module**: This module is in charge of generating a list of agents and static objects which are perceived by the agent in the scene. Working as a synthetic vision, it filters the objects using vision angle and range parameters. It is also in charge of decoding the messages sent by other agents using a dedicated ACL, which can contain verbal communication and information about the other agent's state. In particular, the sender's active role must be included in the message. When the module detects another agent that was not previously detected, it generate a new belief (e.g. "*(Gerard has just been perceived*)"), which will be later removed by the initial plan handling the interaction.

- **Cognitive module**: The cognitive module obtains data from the perception module and encodes it using the assigned role types. Beliefs about other agents are especially important to generate. Ideally, both the perceived entity and its current action are encoded, which allows the designers to create more realistic conversational agents. This task can be simplified by the broadcasting of messages describing the agent activity, when an action is selected. If an entity is recognized, its type and its type's default scores are added to the beliefs database (e.g. "(*Gerard is of type salesman)*", "*(Gerard honesty is low)*"). When the global weight for a social value does not surpass a given threshold, not corresponding beliefs are generated, which eventually inhibits the execution of any differentiated plans.

- **Type identifiers**: In order to assist the cognitive module in linking entity characteristics to types, dedicated type identifier modules can be provided. A general type identifier mechanism which simply links a list of entity IDs to a type is necessary, but a more sophisticated processing could give interesting results. To determine an entity type by processing its known characteristics, a set of rules could be run in the inference engine, as it is done in expert systems. Neural networks are also ideally suited to this task, because of their ability to identify patterns in the data. For instance, perceptrons could be trained to recognize the agents' gender using their sizes and garments.

- **Social reasoner**: This module, which includes a production engine, processes the behaviours preconditions using the beliefs generated by the cognitive module. When a rule is fired, new beliefs are generated or actions are triggered. Standard reasoning based on social or physical data can take place, including planning. However, only a subset of the available behaviours is placed in the inference engine. A score is calculated for each behaviour, which represents the matching degree of the behaviour to the agent's values using the specified weights. All behaviours from the active role, and active roles' parents, whose score does not surpass a certain threshold, are removed from the list of available behaviours. The score itself or a corresponding intensity category is also placed in the belief environment (e.g. "*(plan man:woman_machogreeting social values adequacy is low)*"), which allows the behaviours' preconditions to reference it, for example do define a "reluctant" way of executing a plan. At the end of the process, the selected actions' IDs and parameters are sent to the actions manager.

- **Actions manager**: This module collects the adequate action scripts and runs them successively. Tasks such as keyframe animation of deformable bodies, speech synthesis and rendering, are performed. The actions manager is also in charge of stopping the ongoing actions when a role switch is detected and of queuing the new actions received from the same role.

4.5 Prototype

We have implemented a simplified prototype for SGI workstations. Our implementation takes advantage of the following platforms:

- ANSI C++ and the Standard Template Library (STL),
- the Allegro CL 5.0 Lisp implementation,
- Python language v1.5.2

C++ is used for the reactive mechanisms, animation and rendering but also for perception, since it is very efficient to handle and filter objects pipelines. A C++ class hierarchy has been developed to handle all the necessary data, providing a convenient interface for their manipulation. Information about the library which is in charge of perception can be found in a paper by Bordeux, Boulic and Thalmann (1999).

The Lisp environment is well suited to rules matching and encapsulates a production system. VRLAB's Lisp-based production system has been described by Caicedo and Thalmann (1999).

We are also using a Python interpreter working as a high level interface to our graphics libraries, which allows the easy execution of behaviour scripts without compilation. This interpreter has required the development of an AgentLib++ Python wrapper which links Python to C/C++ libraries (Kallmann et al. 2000). Our standard actions catalogue is mainly based on keyframe animation, with actions such as gestures, postures, walking, etc.

4.5.1 Software Architecture

Figure 42 shows the prototype software architecture, which can be described through a sequence of actions it is in charge of. When it is run, the prototype reads a configuration file including a list of roles and a list of agents to create, with their corresponding role weights. It first initialises the Lisp environment, the graphical environment (including the scene), and creates a window displaying the scene. It also creates the necessary C++ objects (SAgent, SRole, etc.) and integrates all the required data, such as values, in the agent objects according to the roles' specific weights.

Then, the C++ main loop is entered and the perceptive, cognitive and action methods of every declared agent are successively called. The perception method filters the environment and returns a list of perceived entities with their known characteristics, and queued ACL messages are decoded. When cognition is performed, beliefs about these entities are generated. Role switching may also occur during this phase if necessary. Depending on the active role, a number of StandardBehaviours, compatible with the agent's values, are placed in the production engine through C library calls. Finally, forward chaining is run in the inference engine, which checks these behaviour's preconditions, and the script corresponding to the best available option is sent to the Python interpreter. The actions are performed and rendered, before a new perception/cognition/reasoning process is executed.

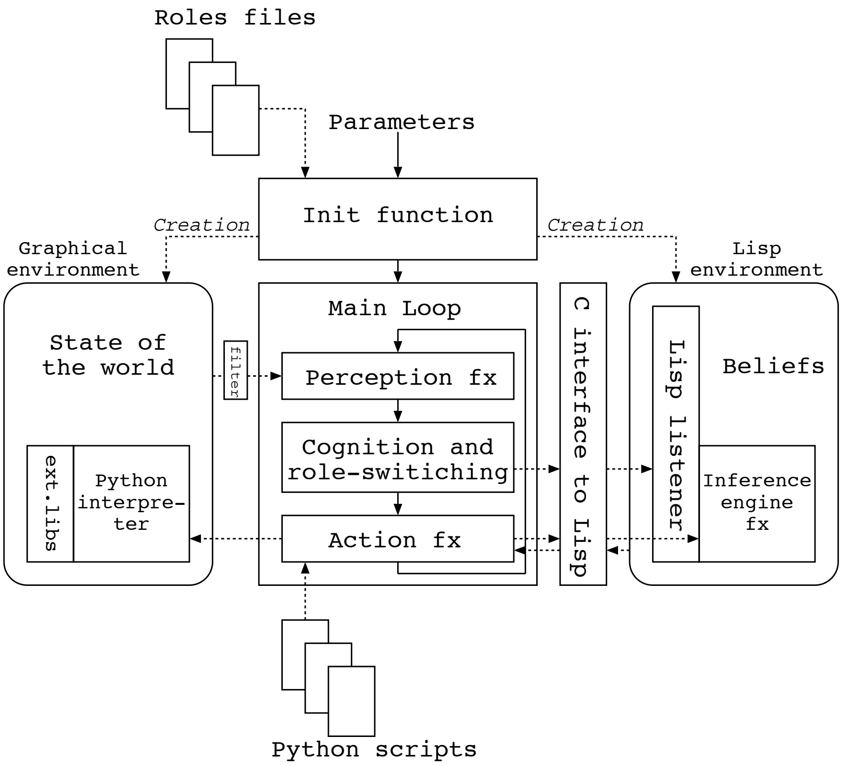


Figure 42 Prototype implementation

It should be mentioned that the integration of multiple platforms and languages has some overhead. In particular, the management of a Lisp environment inside a C++ process required a significant development effort. At the time of the implementation, the C interface to the Allegro platform was the only one available, which complicated to some extent the dynamic generation of the Lisp environment's content. However, the use of the pre-existing C++ components made the development a lot easier, e.g. for perception tasks, and it was a central requirement to reuse VRLAB's Lisp-based reasoning functions.

One of the main differences between the prototype and a full agent implementation of the described mechanisms, is that several agents are here controlled by the same process, which simplified many tasks like message exchange.

4.5.2 Example Scenario

Here is a simple scenario illustrating the behavioural process and basic adaptive behaviour in our prototype implementation. Four social agents are placed in a 3D scene. They have only been assigned a "male role" or a "female role", which inherits data from a general "human role". The behaviours included in the roles are courtesy rules and a general random walking behaviour which is selected by default.

When the simulation starts, our social agent Elvis goes for a walk and will try to appropriately greet the other agents he will meet in the scene. In Figure 43, the presence of another nearby agent has been detected by the perception module. The perception loop is suspended and the cognitive module tries to identify the encountered agent's type. A Lisp-based gender identifier, based on height and garment, is dynamically loaded and run.

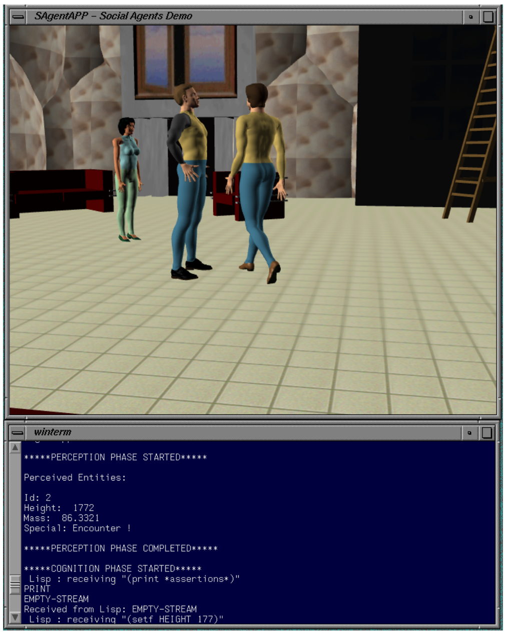


Figure 43 Another agent is perceived

Figure 44 A male agent is greeted

A new belief stating that the encountered agent is a male has been generated in Figure 44. Rules matching is then performed, and a specialized greeting for other men is found in the current active role ("male role"). The corresponding Python script is run.

Figure 45 A female agent is greeted

Figure 46 A neutral greeting is performed

In Figure 45, a female agent is met. The "male role" defines a different way to greet women, which is chosen and performed.

Finally, a third agent is encountered in Figure 46. Because of its height and garment, its type is not recognized and preconditions are not fulfilled for any behaviour in the "male role". However, since the "male role" inherits behaviours from a general "human role", a default action can still be chosen and a gender-neutral greeting is performed.

4.6 Conclusion

The role-based architecture for social agents we have presented in this chapter is only one possible approach of the modelling of *Homo sociologicus*. However, it has the interesting property of allowing the generation of rich social behaviours as the number of specified roles grows: because of the interactions of conflicting norms and values in the multiple roles assigned to an agent and of the manner they are integrated to compose its personality, sequences of actions can be selected in the roles hierarchy in a way that was not planned or anticipated. Like in real life, conflicting roles and motivations may result in the agent's passivity or in the selection of alternative behaviours that can be interpreted as the result of internal tension and/or dilemma.

The main drawback of our approach is that it requires a strong effort in the design of the social roles. However, the parametrization of BDI agents is also a heavy task, and the incremental nature of our approach, which greatly facilitates behaviours reuse, could partly compensate for this work. More generally, we think that there is no way to avoid such a behaviour specification effort for the simulation of culturally situated human behaviours. Techniques like neural networks and genetic algorithms, which minimize explicit behaviour specification, have been successfully used to generate simple social behaviours, as in the work by Channon and Damper (1998), but are not well suited to the production of social behaviours that are specific to a cultural context and are the result of a complex historical process. For example, it would be extremely problematic, in the current state of the art in both computer science and social sciences, and because of the complexity of the required fitness function, to simulate the behaviour of a passer-by, in Lausanne, at the beginning of the third millenium, by simply evolving a population of generic agents using some basic constraints. To simulate culturally situated behaviours, we argue that it is necessary to rely on specifications given, in one form or another, by a human observer which is immersed in the reality to model.

The usefulness of integrating such concepts as social values and norms in an agent architecture may also be questioned. We think that this task has a clear heuristic value for the simulation of social phenomena and could help achieve a better comprehension of individual's behaviours. In addition, as has been described in the preceding chapter, when human users interact with an embodied agent, they have a number of conscious and unconscious social expectations relative to its behaviour. When these expectations are met using norm-based behaviours, the quality of the users' experience is greatly improved. We also think that the integration of social values in agent systems can be extremely valuable when a choice has to be made in trade-off situations, especially for life-critical systems. For example, in the case of an agent directing a railway system, situations may occur where productivity, the main agent's "desire", has to be sacrificed because humans are potentially endangered, a situation related to a higher-order values system. Actually, Asimov's laws of robotics (1968) can be seen as the anticipation of an hard-coded implementation of such a value system into robots. These laws "act" independently from the standard goals assigned to a robot to constrain its selection of actions. The difference with our approach is that we work with the sociological definition of the social value concept which is more general than what is commonly referred to as "morality", and do not specify the values to implement, since this choice is fundamentally a matter of cultural preference (e.g. translated into self-questioning, motivation is not restricted to "what's right" or "what preserves human life" but may include "what's cool", "what's fashionable", "what's lucrative", "what's politically correct", etc., depending on the socialization process). Another important point is that we explicitly link behaviours to social value ratings, whereas many existing systems contain an implicit social values system directly hard-coded in their functioning rules. This feature would make control systems easily and dynamically adjustable to the evolving societal requirements.

In this chapter, we have discussed the paradigmatic opposition between the *Homo economicus* and *Homo sociologicus* models. Although the two dimensions are present in our approach, *Homo sociologicus* has high precedence. It conditions the agent goals and beliefs, and then, the selection of action inside a restricted framework can be computed using any technique, including the calculation of costs and benefits (except for norm-based behaviour). However, an extremely interesting goal for future developments would be to make both apparatus interact more closely. A strong theoretical basis is important for this project since the temptation is great to dissolve one model into the other, rather than really integrate them. For instance, a possible approach would be to consider that an agent can choose to fulfil a social norm or not, depending on its own interests and evaluation of the social consequences. Although it seems compatible with the *Homo sociologicus* paradigm, such a conception does not take into account the individual's internalisation of the societal norms and values, i.e. they become a part of himself/herself rather than constituting an external constraint. The same critic applies to approaches which conceive benevolent agents solely as self-interested in the long-term.

For the time being, the theoretical base is unfortunately weak, even if some researchers tried to bridge the gap between the two orientations. Parsons (1951a) himself, in his theory of action, explains the individual's action through both social values and personal interests, and combines the orientation of the action toward an end with norm-based behaviours. More recently, Anthony Giddens (1984) tried to unify both orientations in his "theory of structuration". Though, it remains to be seen whether the current state of these theories' formulation would allow the implementation of a corresponding agent architecture. An additional challenge would be to retain the important properties originating from the coupling of an agent with its environment (Dautenhahn 1999) while integrating such a model in an embodied agent.

Our approach of the modelling of social agents, presented in this chapter, can be summarized as follows: a social agent based on the *Homo sociologicus* model is made up of a set of social roles, and its "freedom", defined as its behaviour's unpredictability, emerges from the contradictions between these roles. We have only scratched the surface of the issue of social agents modelling, but we hope that our work can help introduce a somewhat different perspective for future research. In particular, the proposed conceptual architecture may prove useful in the framework of the simulation of face-to-face interactions.

In the preceding sections, our research was focused on the modelling of complex individual social beings, including its cognitive and symbolic reasoning aspects. We are now going to tackle the issue of the simulation of group behaviour and group dynamics, and, for this purpose, we will explore a different approach, based on simple but strongly interrelated reactive agents.

Chapter 5 ● Simulation of Small Groups Dynamics

5.1 Introduction

Group dynamics, as it has been defined in the introduction, is a complex social phenomenon whose very nature must be properly taken into account for the realistic simulation of social interaction in small groups. The fundamental observation that group dynamics emerges from the interactivity of the group members and, in return, affects the regularities in their interactivity, leads to the identification of two types of mechanisms that must be provided:

- a mechanism simulating the influence that group members exert on each other in the action selection process,
- a mechanism simulating the transformation of group members' interpersonal relationships triggered by their interactions

In addition, a third type of mechanism can be included:

- a mechanism simulating the long-term impact on group member's interaction rates induced by the group's functional requirements

We argue that the coupling of such three mechanisms allows for the realistic simulation of the generation and organization of face-to-face interactions in small groups as an interactive and evolving process.

In order to simulate the influence that participants exert on one another, we propose in this chapter a specific action selection mechanism in which a flow of activation energy circulates between group members, who can contribute to activate or inhibit others' actions. Our model is mainly based on the observations and empirical data gathered by Bales (1951) in his sociological study of small groups. In a different perspective, psychologists have been studying for a long time interpersonal relationship development and have proposed several models and classifications. We choose Knapp's (1996) relational stages model as a basis for our implementation of a mechanism triggering changes in interpersonal relationships in the context of small groups. Our relationship development model is based on the premise that a series of successful interactions, engaged over time, leads to an emotional response specific to the interaction type, which in turn can trigger a structural change in the interpersonal relationship. This view is based on recent work in the theory of relational cohesion from an exchange perspective (Burke 2003). In order to develop the third type of mechanism, we exploit the Tuckman (1965) model of group forming.

A consequence of simulating small groups, where members have a direct relationship to each other and interact in face-to-face situations, is that it is needed to take into account the individual social characteristics of the members (Bales 1970) and the specificity of their relationships. Although the precise modelling of individual traits is not as important in the context of small groups as it was in the preceding chapter, these still have a significant impact on the resulting behaviours compared to what happens, for example, in a crowd context. Existing simulated social networks using graph based models, like in the work by Stocker, Green and Newth (2001), have been successfully used in order to identify mechanisms acting on the scale of large groups, but are not dedicated to the fine modelling of specific micro-social contexts. For small groups, a more detailed control of the connection between nodes is required, with the ability to provide a connection probability relative to a specific type of interaction, allowing the superimposing of several fully or partially connected graphs. In order to allow for the social characterization of group members and for the convenient specification of their relationships, we present in this chapter the Social Identity Markup Language (SIML), an XML-based language for the specification of agent social identity.

As just discussed, our work is related to research in simulated social networks, as well as to a number of approaches which have been presented in the literature review. In particular, it has common features with Carley's (1991) model of group stability, since in her model like in ours, agents interact with a probability proportional to their relative similarity. However, in contrast to our approach, "similarity" is, in her work, solely defined in terms of shared knowledge. Our contribution is also related to approaches inspired from the field of ethology, like Tyrrel's (1993), as well as agent architectures trying to solve the action selection problem, e.g. Maes's (1989), since the mechanism we propose for the generation of interactions is based on a algorithm of activation and inhibition energy triggering actions when a given energy threshold is reached. An interesting difference is that, in our approach, energy is not only spread inside one organism but its distribution is realized by all group members together.

In addition to SIML and to a model of small groups dynamics and interpersonal relationship development, we present in this chapter a tool for the control and visualization of group interactions called 3DSociogramViewer. Indeed, we think that it is particularly important to allow simulation designers to be able to visualize the group's internal processes and to introduce perturbation in the system, be it for the exploration and study of small groups or for purely simulation-oriented applications.

The initial section of this chapter is dedicated to a short presentation of the sociological framework we chose for our work on small groups.

5.2 Sociological Framework: Bales's Interaction Process Analysis

Interaction Process Analysis (IPA) (Bales 1951; Bales 1970) is a well known methodology for the observation and analysis of small groups developed by Robert F. Bales in the Social Interaction Laboratory at Harvard University. It is especially well suited to serve as a basis for a system simulating small groups dynamics since it has the following unique advantages:

- It defines a standard set of categories used to code social interactions taking place in the context of small groups.

- A significant amount of standardized empirical data describing existing groups is available.

The categories are of general-purpose, rather than dedicated to a particular kind of group or context, and they are derived as clearly as possible from a generalized theoretical framework. They are also meant to be fully inclusive, and every observed action should match one and only one category. In addition, detailed instructions and training methods for each category are provided in order to help observers achieve a high degree of reliability and reproducibility of the coded data.

Participants' verbal and nonverbal behaviours are first broken down into acts or micro-interactions, e.g. saying a sentence or performing a gesture, which are coded into one of the 12 standardized categories described in the next section. The interaction initiator and recipients are designated by an assigned number which is positive (e.g. "3-1" represents an interaction initiated by group member 3 with group member 1), while the in-group as a whole is designated by a zero (e.g. "3-0" represents an interaction of group member 3 with all other group members). The observation lasts for a duration that has been previously decided and can take place in a laboratory setting as well as in the participants' natural environment. Groups' and group members' profiles are then established on the basis of interaction rates and can be safely analysed using statistical tools, compared with other profiles, or used as input for a dedicated simulation system.

We work with the original IPA categories and not the 1970 revised versions (Bales 1970), because they are commonly used and much more empirical data is available. The more recent System of Multiple Level Observation of Groups (SYMLOG) (Bales 1980), with its 3 dimensions and 26 categories, was another interesting candidate framework. However, since it is used to rate group members' personality on a long timeframe rather than characterize atomic interactions, it is not as well suited to our simulation application.

5.2.1 IPA Categories

Table 5 gives an overview of the 12 IPA categories. A full description of the associated behaviours is provided by Bales (1951). The categories are divided into four groups: Section A defines three Positive Reactions, section B constitutes a group of Attempted Answers, section C contains activities which can be characterized as Questions, and section D defines Negative Reactions. More generally, section B and C constitute an area of Task Problems, while sections A and D constitute an area of Social-Emotional Problems.

Section	IPA category	Description
A: Social-emotional/Positive	**1. Shows solidarity**	Raises other's status, gives help, rewards
	2. Shows tension release	Jokes, laughs, shows satisfaction
	3. Agrees	Shows passive acceptance, understands, concurs, complies
B: Task-oriented/Attempted answers	**4. Gives suggestion**	Gives direction, implying autonomy for other
	5. Gives opinion	Gives evaluation, analysis, expresses feeling, wish
	6. Gives orientation	Gives information, repeats, clarifies, confirms
C: Task-oriented/Questions	**7. Asks for orientation**	Asks for information, repetition, confirmation
	8. Asks for opinion	Asks for evaluation, analysis, expression of feeling

	9. Asks for suggestion	Asks for direction, possible ways of action
D: Social-emotional/Negative	**10. Disagrees**	Shows passive rejection, formality, withholds help
	11. Shows tension	Asks for help, withdraws out of field
	12. Shows antagonism	Deflates other's status, defends or asserts self

Table 5 IPA categories

A symmetrical relation exists between the top half and the bottom half of the list of categories, as each pair of categories is concerned with a particular aspect of the interactional process. Although every type of interaction may be found occurring at any point in time, the categories have been organized in order to emphasize the emergence of typical action/reaction sequences.

5.2.2 Empirical Regularities in Small Group Interactions

In addition to providing the IPA methodology, Bales observed and documented a number of very interesting empirical regularities about small groups. He was particularly interested in identifying the typical chains of actions and reactions that occur in this particular context. The tendency toward repetition of some acts and the tendency toward alternation of others is a classic example. Bale's findings are considered to be quite general and to hold within a wide range of conditions.

The first chains of actions and reactions identified by Bales are relatively basic but have been validated by IPA research. Questions (cat. 7, 8, 9) tend to elicit answers (cat. 4, 5, 6), answers or task attempts (cat. 4, 5, 6) tend to elicit agreement or disagreement (cat. 3, 10) as a closure of the sequence, friendly acts (cat. 1, 2) tend to elicit friendly acts and unfriendly acts (cat. 11, 12) tend to elicit unfriendly acts (Bales 1970). Here is how the observed regularities are described in Bales's work (1951):

- There is a tendency toward repetition for categories 2, 4, 5, 6 and possibly 9: a repetition in the same category is the most subsequent act.
- There is a tendency for acts of high general frequency to appear frequently as subsequent act for all categories considered as prior acts: for cat. 1, 3, 7, 8, 10, 11, 12, considered as prior acts, cat. 5 or 6 is the most frequent subsequent act.
- There is a tendency for categories defined as questions to lead to categories defined as answers: cat. 7 leads to cat. 6, cat. 8 leads to cat. 5, cat. 9 leads to cat. 4.
- There is a tendency for categories defined as answers to lead to agreement and disagreement act: for cat. 4, 5, 6, cat. 3 and cat. 10 are the most subsequent act.
- There is a tendency for categories 10, 11, 12 to lead to their opposite categories 1, 2, 3: except for cat. 11 and cat. 12 leading to cat. 2, the connection is however weak.

The most fundamental tendency observed by Bales is that of an alternating emphasis on two main types of activities: "When attention is given to the task (cat. 4-9), strains are created in the social and emotional relations of the members of the group, and attention then turns to the solution of these problems. So long as the group devotes its activity simply to social-emotional activity (cat. 1-3, 10-12), however, the task is not getting done, and attention would be expected to turn again to the task area." (Bales 1951, p. 8).

In Bales's functionalist view, these two type of activities correspond to fundamental social functions that must be fulfilled in order for the group to maintain a kind of equilibrium and thus be able to survive. In his perspective, the successful transition through any particular phase, in a more or less regular pattern, is necessary to the maintenance of the interaction system. The emerging distribution of phases in the time dimension and between group members constitutes what Bales calls an "agenda topics".

5.2.3 Group Member Motivation

Although Bale's research started prior to the explicit definition of the "social" as a phenomenon emerging from the participant's interactivity, his conception of social systems is consistent with this modern view: "There's a system-influence which is distributed between members, so that one discovers the pattern of the system only by looking at the total activity of all members put together over the total time or, in other terminology, by looking at the social system and not simply at the individual roles." (Bales 1951, p. 58). Interestingly, his conception of social systems also includes a feedback loop: "Although the social structure of the group and its culture both arise out of interaction and are formed by it, once formed, they constitute a part of the framework within which further interaction proceeds. In a similar way the personality arises out of interaction and is formed by it, but once formed, in however small a respect, it becomes a part of the framework within which further changes take place." (Bales 1951, p. 66)

Bales's conception of group member motivation is related to this idea that the total process of action in itself becomes a context within which a given act takes place: "“Motivation” is a term we often use to designate the subjective combination within a given individual of these various structured influences on interaction. [..] In this sense, motivation can be regarded as the total state of the individual in relation to his total situation which is presumed to exist before and during any given act and which impels him to do what he is doing." (Bales 1951, p. 66). In other words, group members have the ability and natural tendency to choose actions which are logically relevant to what has happened before and what they expect to come next. With regards to time involvement, they proceed on a cyclic basis from a state of tension and heightened motivation toward tension/motivation reduction, and back to a period of tension/motivation accumulation.

Since the following of an adequate "agenda topics" results in the system's stability, and since all members have some interest in bringing about stability, they naturally collaborate in order to address the group's functional requirements. The individual's tensions resulting from the uncertainty and unpredictability of other's action is controlled this way.

Bales's work has been extremely important in the developments we present in this chapter. His conception of group member motivation inspired the design of our action selection mechanism, while the empirical regularities he observed constitute its main substance, i.e. the rules generating reactive activation and inhibition energy have been written on the basis of these observations. The IPA categories themselves play a central part in both the group dynamics engine and the relationship development model.

5.3 Social Identity Markup Language

SIML is an XML tag-based language allowing simulation designers to easily define groups and group members, more generally, agents' social identities. It has been designed general enough to be used in different applications, and includes a number of standard variables commonly found in sociological and psycho-sociological research in addition to IPA interaction rates. These variables have been chosen because their impact on interpersonal behaviour has been widely studied and documented. They will be shortly presented but an extensive discussion of their characteristics and relative influence is beyond the scope of this document.

In our work, SIML is used to convey IPA interaction rates to the simulation system, but also for the computation of the proactive dimension of the member's activity, i.e. its tendency to communicate with other group members according to their individual traits and independently of the interactive context.

5.3.1 Syntax

Three main layers of information are present in the language (see Figure 47): (1) general social-statistical and personality characteristics, (2) information about the agent's roles and status within the belonging groups, and (3) interpersonal relationships and interaction data.

The social-statistical variables - gender, age group, culture - used in the first part of the <*SocialAgent*> data structure, are classic and have been used in numerous sociological studies. Selecting the options to propose was straightforward for the gender setting, and, for the age group setting, we considered that four categories (child, adolescent, adult, elder) were sufficient for most applications. However, the suitable choice of cultural belongings is dependant on the nature of the phenomenon to be simulated. By default we propose the "geographic realms" (Western Europe, Russia/Eastern Europe, North America, Middle America, South America, North Africa/Southwest Asia, Sub-Saharan Africa, East Asia, Southeast Asia, Australia, Pacific Realm), which are well known in the field of cultural geography and take into account, in addition to the geographic criterion, the issue of social identity; nevertheless, it is clear that their degree of granularity is not sufficient for many applications. The personality settings that follow - extraversion, agreeableness, conscientiousness, emotional stability, intelligence - are based on the so-called "big five dimensions of personality" identified and used by many researchers, for example in the study of group leaders emergence (Digman 1990). The following expressions are used to characterize these dimensions (Forsyth 1999, p. 351):

- extraversion: outgoing, sociable, interpersonal, expressive, gregarious,
- agreeableness: friendly, warm, likable, generous, kind,
- conscientiousness: responsible, achievement oriented, dependable, self-controlled,
- stability: emotionally controlled, assured, not anxious, balanced,
- intelligence: intellectually able, open to new ideas and experiences, cultured

The second layer of information is specific to the groups that the agent being modelled is member of, and is signalled by a number of <*Group*> tags. Three status-related variables -

leadership, power, prestige - commonly distinguished in small groups literature, are included in this section. When power is about influencing, guiding and controlling other's action by asking or coercion, prestige is not linked to resource or participants control, but typically to the achievement of valued goals, as noted by Bales (1951). Leadership is generally considered as a special form of power, in which a small number of individuals act in the best interests of a group, with the consent of that group, and by distributing some of their power to other members (Hollander & Offermann 1990).

```
<SIML>
<SocialAgent>
<Name>string "name of agent"</Name>
<Gender>string Female,Male</Gender>
<AgeGroup>string Child,Adolescent,Adult,Elder</AgeGroup>
<Culture>string WesternEurope,Russia/EasternEurope,NorthAmerica,MiddleAmerica,SouthAmerica,
NorthAfrica/SouthwestAsia,SubSaharanAfrica,EastAsia,SoutheastAsia,Australia,PacificRealm</Culture>
<Extraversion>float 0.0 <= f <= 1.0</Extraversion>
<Agreeableness>float 0.0 <= f <= 1.0</Agreeableness>
<Conscientiousness>float 0.0 <= f <= 1.0</Conscientiousness>
<Stability>float 0.0 <= f <= 1.0</Stability>
<Intelligence>float 0.0 <= f <= 1.0</Intelligence>
<Group Id="integer 1 <= i" - optional>
  <GroupName>string "name of group" - optional</GroupName>
  <IdInGroup>integer 1 <= i</IdInGroup>
  <Role>string "name of role" - multiple</Role>
  <Leadership>float 0.0 <= f <= 1.0</Leadership>
  <Power>float 0.0 <= f <= 1.0</Power>
  <Prestige>float 0.0 <= f <= 1.0</Prestige>
  <Relationship With="integer 0 <= i" - optional>
   <Liking>float 0.0 <= f <= 1.0</Liking>
   <Familiarity>float 0.0 <= f <= 1.0</Familiarity>
   <Trust>float 0.0 <= f <= 1.0</Trust>
   <Commitment>float 0.0 <= f <= 1.0</Commitment>
   <InteractionRates – optional>
     <ShowSolidarity>float 0.0 <= f <= 1.0</ShowSolidarity>
     <ShowTensionRelease>float 0.0 <= f <= 1.0</ShowTensionRelease>
     <Agree>float 0.0 <= f <= 1.0</Agree>
     <GiveSuggestion>float 0.0 <= f <= 1.0</GiveSuggestion>
     <GiveOpinion>float 0.0 <= f <= 1.0</GiveOpinion>
     <GiveOrientation>float 0.0 <= f <= 1.0</GiveOrientation>
     <AskForOrientation>float 0.0 <= f <= 1.0</AskForOrientation>
     <AskForOpinion>float 0.0 <= f <= 1.0</AskForOpinion>
     <AskForSuggestion>float 0.0 <= f <= 1.0</AskForSuggestion>
     <Disagree>float 0.0 <= f <= 1.0</Disagree>
     <ShowTension>float 0.0 <= f <= 1.0</ShowTension>
     <ShowAntagonism>float 0.0 <= f <= 1.0</ShowAntagonism>
   </InteractionRates>
   <InteractionsAverageSuccess – optional>
     <ShowSolidarity float 0.0 <= f <= 1.0</ShowSolidarity>
     ...
     <ShowAntagonism>float 0.0 <= f <= 1.0</ShowAntagonism>
   </InteractionsAverageSuccess>
  </Relationship>
</Group>
</SocialAgent>
<!-- ... several agents can be defined in the same .siml file -->
</SIML>
```

Figure 47 The SIML agent syntax

In addition to a social status, a number of social roles can be assigned to each agent in a group. We have decided that the definition of some general roles must be provided by the SIML parser itself (e.g. a role named *NewMember* which typically contains low interaction rates) but that other more specific ones could be defined by the designers at the root of SIML files (see Figure 48), which is also a way to facilitate code reuse. One or several references to fulfilled roles can be introduced inside a <*SocialAgent*> section. Following Bales's view, a role is defined as a profile of interaction rates with all other group members (signalled with "<*Relationship With=**>"), or with each one individually. When several distinct roles are assigned to one agent, the parser will automatically compute a set of average rates for each interaction type with each member.

```
<SIML>
<SocialRole Name="string "name of role"" GroupId="integer 1 <= i">
  <Relationship With="integer 0 <= i or *">
   <InteractionRates>
     <ShowSolidarity>float 0.0 <= f <= 1.0</ShowSolidarity>
     ...
     <ShowAntagonism>float 0.0 <= f <= 1.0</ShowAntagonism>
   </InteractionRates>
   <InteractionAverageSuccess – optional>
     <ShowSolidarity>float 0.0 <= f <= 1.0</ShowSolidarity>
     ...
     <ShowAntagonism>float 0.0 <= f <= 1.0</ShowAntagonism>
   </InteractionAverageSuccess>
  </Relationship>
</SocialRole>
<!-- ... several agents and roles can be defined in the same .siml file -->
</SIML>
```

Figure 48 The SIML role syntax

The relationship of the agent with other group members is defined in the <*Relationship*> tag using four variables documented by, among others, Knapp (1996): Liking, Familiarity, Trust and Commitment. These variables together form an intimacy profile which is specific to each relationship. Their relative influence will be discussed together with the description of our relationship development model.

The <*InteractionRates*> tag follows but is still part of <*Relationship*> tag. As previously explained, the interaction rates with every other group members, which are defined in this section, are based on IPA categories. Unfortunately, Bales did not provide a specific unit for its IPA interaction rates[5], thus we had to choose one.

Considering a standard act duration of 10 seconds, an interaction rate $r \in \mathbf{R}$: 0.0 <= r <= 1.0, and a count of performed interactions during 1 hour $c \in \mathbf{N}$:

$$r = \frac{c}{360}$$

[5] The number of acts and the duration of the observation are normally mentioned, but this duration, typically 60 or 90 minutes, varies from one observation to another.

For example, with r=1.0 and letting aside random variation and any external influence, a new interaction of the specified type will be triggered every 10 seconds. Each interaction rate can be complemented with an average interaction success, which ensures that a specific level of influence on relationship development is met on the long term, in spite of continuous small random variations. In addition to standard dyadic interactions, multilateral interactions involving all group members can also be specified (in accordance with Bales's notation using "*<Relationship With=0>*").

As many <*SocialAgent*> and <*SocialRole*> tags as required can be specified at the root of the document. A SIML document may define all members of a particular group, only one or some members of that group, as well as members of different groups, depending on the designers' preference. Since only the first information layer (social-statistical and personality characteristics) is mandatory in the language definition, SIML is well suited to be used in different applications.

5.3.2 Parser Implementation

Since SIML is based on XML, the parsing but also the streaming and validation of the data are greatly facilitated. The implementation of our parser makes use of the standard DOM model (*DOM homepage*) and of its associated Windows libraries. It has been preferred to the SAX model (*SAX homepage*), since it makes it easier, thanks to the tree structure it generates, to write back to disk a modified version of the SIML document, typically representing the new state of a group after some simulation iterations have taken place. The SIML Schema (*XML Schema homepage*) allowing the straightforward validation of the SIML documents has been written, which could also prove useful in the dissemination of the format.

In addition to the loading and saving of SIML documents, our parser is able to merge different SIML files together. When several groups are referenced in a single .siml file, the parser must first obtain (from the user, through the 3DSociogramViewer application) the ID of the group to process. Then, it is able to load additional .siml files, extract from these files the relevant group member profiles, and update the tree structure in memory.

An important aspect of SIML is the flexibility provided to designers, who can choose the level of detail they need in the simulation parameters. If basic profiles only are provided, a statistical approximation of the dyadic interaction rates will be computed by the parser using a set of dedicated rules, described in the next section, which allows for a quick and easy setup of the simulation. The use of predefined roles and design of a set of specific new ones matching as well as possible the members of an existing group, is another possibility which can give more realistic results depending on the situation. To code real-life observations and to include in the script detailed interaction rates is the best solution, but it requires a more important design effort. Our SIML parser allows the three strategies to be freely combined.

The correct blending of the interaction rates directly provided in the <*SocialAgent*> data structure with the ones originating from several <*SocialRole*> tags, is done by computing the weighted average of all rates related to a pair of (directed) relationship and interaction types.

5.3.3 Default Dyadic Interaction Rates

When dyadic interaction rates are not provided in the SIML file, we use a number of rules inspired from statistical group studies, for example Hallinan's (1981), to compute default rates based on individual traits. A description of these rules follows:

1) The most general rule is called "homophily" and states that similar members tend to regroup in clusters, where very high interaction rates are observed. They also tend to like each other and express it through verbal and nonverbal messages on a regular basis.

- A similarity index based on (1) age, (2) gender, (3) ethnicity (SIML's culture setting), (4) personality (extraversion, agreeableness, conscientiousness, emotional stability, intelligence) and (5) status (leadership, power, prestige), is calculated.
- A multiplying factor specific to each IPA interaction - higher for the Social-emotional/positive types - is set.
- Basic interaction profiles are generated by multiplying both values, which provides higher overall and Social-emotional/positive interaction rates for similar dyads.

2) A second set of rules makes use of personality traits and of their documented effects on interaction rates:

- The "Extraversion" score provides a secondary activity reference level and is applied to each interaction rate of the person.
- "Agreeableness" and "Stability" are traits that are generally looked for in small groups and are therefore used to fine-tune the desire of one member to interact with another.

3) Additional rules have a more specific influence on interaction rates:

- Leaders have relative high rates in "Gives orientation" and low rates in "Asks for orientation" / inverse proportion for members with a low Leadership rating.
- High power members have relative high rates in "Gives suggestion" and low rates in "Asks for suggestion" / inverse proportion for members with a low Power rating.
- High prestige members have relative high rates in "Gives opinion" and low rates in "Asks for opinion" / inverse proportion for members with a low Prestige rating.

We think that the SIML syntax and its associated parser features constitute a powerful tool for social simulation. It serves as a foundation for the developments we are presenting in the next sections, but could also prove valuable in different contexts.

5.4 A Model for the Generation and Organization of Small Groups Interactions

As previously explained, our model of group dynamics is dedicated to the simulation of the stream of face-to-face interactions that is observed in small groups. The goal is to generate, in a given period of time, for each group member with its specific social characteristics, an adequate number of interactions of each IPA type. The coherence of the organization of these interactions in the temporal dimension is another important aspect of the simulation. In order to model group members' individual motivation and behaviour resulting in emerging group dynamics phenomena, the presented mechanism deals with all combinations of possible interactants/interaction types as competing and interlinked behaviours.

We consider two dimensions in the problem of action selection applied to small groups: (1) a proactive dimension, (2) a reactive dimension.

The proactive dimension is linked to each group member's motivation to interact with the others based on their personal characteristics. The reactive dimension refers to the immediate influence group members exert on each other's behaviour. Instead of classifying participants' acts either as pure proactions, e.g. ask for opinion, or reactions, e.g. give opinion, we combine both dimensions in the group member motivational apparatus. This is consistent with empirical observations (e.g. it is frequent to give one's opinion even when not asked) and allows for opportunistic behaviours to be triggered (illustrated with the sentence "I'm happy that you mention it, I wanted to tell you about it for some time"). We represent proactive and reactive motivations as amounts of activation energy which can be freely blended.

5.4.1 Proactive Motivation

The computation of the proactive motivation is, in our approach, dependant on dyadic interaction rates provided by the simulation designers or calculated by the SIML parser. Every timestep, a specific amount of proactive energy $p \in \mathbf{R}_+$, which is proportional to the interaction rate r, is added to a threshold automaton. Given:

- another group member index $m \in \mathbf{Z}_+$,
- an IPA interaction type index $i \in \mathbf{Z}$: $1 <= i <= 12$,
- a global parameter representing the simulation timescale $\tau \in \mathbf{R}_+$

$$p_{mi} = \frac{r_{mi}}{\tau}$$

Here is the calculation of the amount of proactive energy including random variations $P \in \mathbf{R}+$, for an interaction type with a specific group member, at a specific timestep, given:

- the remaining proactive energy from processings preceding the last time a *mi* interaction was triggered $Q \in \mathbf{R}$,
- a timestep index $t \in \mathbf{Z}_+$,
- the number of timesteps passed since the preceding *mi* interaction was triggered $n \in \mathbf{Z}_+$,
- a global parameter representing the maximum amount a random activation energy that can be added/subtracted each timestep $\rho \in \mathbf{R}_+$,
- a random weight updated on a fixed interval $w \in \mathbf{R}$: $0.0 <= w <= 1.0$

$$P_{mi} = Q_{mi} + \sum_{t=1}^{n} p_{mi} + 2\rho(w_t - 0.5)$$

5.4.2 Reactive Motivation

Reactive energy is generated by a set of rules adapted from Bales's observations, which are described in Table 6. When an interaction ends, the rules' preconditions, based mainly on the interaction type and participant's role (initiator or receiver), are checked and a subset is executed adding energy to a distinct storage entity. The amount of reactive energy is dependant on the interaction success, which can be directly provided in the SIML file or is computed using a separate set of rules (e.g. "Gives orientation" could be defined as more successful when performed by high status members, the rules we have implemented are specified in Section 5.7.2). A negative amount of reactive energy can also be produced, which temporarily inhibits the selection of a behaviour instead of favouring it. Here is the calculation of the amount of reactive energy $R \in \mathbf{R}$, for an interaction type with a specific group member, at a given timestep, given:

- the remaining reactive energy from preceding processings $S \in \mathbf{R}$,
- a rule index $z \in \mathbf{Z}$**:** $1 <= z <= 20$,
- a global parameter used to set the reactivity of the system $\varpi \in \mathbf{R}_+$,
- the interaction's success $s \in \mathbf{R}$: $0.0 <= s <= 1.0$,
- a function *F(m,i,z)* returning an amount of reactive energy ∈ R: -1.0 <= r <= 1.0 based on the rules in Table 6 (0.0 if the rule is not applicable, i.e. no interaction ended in the current time frame, or if its preconditions are not fulfilled)

$$R_{mi} = S_{mi} + \sum_{z=1}^{20} \varpi s_{mi} F(m,i,z)$$

In order to maintain the global amount of energy in the system, the reactive energy is converted from proactive energy, i.e. an amount of proactive energy equal to the total of affected reactive energy must be removed. Given:

- the number of IPA interaction types 12,
- the number of group members $n \in \mathbf{Z}_+$

$$\textit{for every m,i:} \quad P_{mi} = P_{mi} - \frac{\sum_m \sum_i \sum_{z=1}^{15} \varpi s_{mi} F(m,i,z)}{12(n-1)}$$

Immediacy being the main attribute of reactive behaviours, it is necessary that the influence of preceding interactions decreases in time. For this reason, we have included an automatic decay of reactive energy which occurs every timestep when reactive energy is present. Given:

- $R_{mi} > 0$,
- a global parameter which defines the reactive energy decay speed $\delta \in \mathbf{R}_+$

$$\textit{for every m,i:} \quad R_{mi} = R_{mi} - \delta$$

Given:

- $R_{mi}<0$

$$\text{for every } m,i: \quad R_{mi} = R_{mi} + \delta$$

Again, this energy has to be transferred back to the proactive energy storage entity. Given:

- the sum of all activation and inhibition energy removed in the current time frame by the decay function $D \in \mathbf{R}$

$$\text{for every } m,i: \quad P_{mi} = P_{mi} + \frac{D}{12(n-1)}$$

The following rules, which define the immediate influence that participants exert on one another through social interactions, are based on Bales's observations (1951). Used in conjunction with the presented action selection mechanism, they induce the construction of typical chains of interactions based, among others, on the reciprocity principle. In our implementation, these rules are placed in a separate dynamic library and thus can be easily updated and enriched.

Given:
- an interaction initiator I,
- an interaction receiver R,
- a successful interaction defined as $s \in \mathbf{R}$: $0.5 <= s <= 1.0$,
- an unsuccessful interaction defined as $s \in \mathbf{R}$: $0.0 <= s < 0.5$,
- an arbitrary amount of activation energy $a \in \mathbf{R}$.

General rules:
1. Any I→R interaction increases motivation for a R→I interaction by a and increases motivation for a new I→R interaction by $\frac{a}{2}$
2. Any Social-emotional/positive I→R interaction increases motivation for every Social-emotional/positive R→* interaction by a and increases motivation for every Social-emotional/positive I→* interaction by $\frac{a}{2}$
3. Any Social-emotional/negative I→R interaction increases motivation for every Social-emotional/negative R→* interaction by a and increases motivation for every Social-emotional/negative I→* interaction by $\frac{a}{2}$
4. Any Task-oriented I→R interaction increases motivation for every Task-oriented R→* interaction by a and increases motivation for every Task-oriented I→* interaction by $\frac{a}{2}$

Social-emotional interactions specific rules:
5. A successful "Shows solidarity" I→R interaction increases motivation for a "Shows solidarity" R→I interaction by a, decreases motivation for a "Shows antagonism" I→R interaction by a, and decreases motivation for a "Shows antagonism" R→I interaction by a
6. A successful "Shows tension release" I→R interaction increases motivation for a "Shows tension release" R→I interaction by a, decreases motivation for a "Shows tension" I→R interaction by a, and decreases motivation for a "Shows tension" R→I interaction by a

7. A successful "Agrees" I→R interaction increases motivation for a "Agrees" R→I interaction by a, decreases motivation for a "Disagrees" I→R interaction by a, and decreases motivation for a "Disagrees" R→I interaction by a
8. A successful "Shows antagonism" I→R interaction increases motivation for a "Shows antagonism" R→I interaction by a, decreases motivation for a "Shows solidarity" I→R interaction by a, and decreases motivation for a "Shows solidarity" R→I interaction by a
9. A successful "Shows tension" I→R interaction increases motivation for a "Shows tension" R→I interaction by a and decreases motivation for a "Shows solidarity" I→R interaction by a *(not for "Shows solidarity" R→I which matches a possible sequence)*
10. An unsuccessful "Shows tension" I→R interaction increases motivation for a "Shows tension release" R→I interaction by a and increases motivation for a "Shows tension release" I→R interaction by $\frac{a}{2}$ *(mechanism to interrupt the tension escalating loop)*
11. A successful "Disagrees" I→R interaction increases motivation for a "Disagrees" R→I interaction by a, decreases motivation for a "Agrees" I→R interaction by a, and decreases motivation for a "Agrees" R→I interaction by a

Task-oriented interactions specific rules:

12. A successful "Asks for orientation" I→R interaction increases motivation for a "Gives orientation" R→I interaction by a
13. A successful "Asks for opinion" I→R interaction increases motivation for a "Gives opinion" R→I interaction by a
14. A successful "Asks for suggestion" I→R interaction increases motivation for a "Gives suggestions" R→I interaction by a
15. A successful "Gives orientation" I→R interaction increases motivation for a "Agrees" R→I interaction by a
16. An unsuccessful "Gives orientation" I→R interaction increases motivation for a "Disagrees" R→I interaction by a
17. A successful "Gives opinion" I→R interaction increases motivation for a "Agrees" R→I interaction by a
18. An unsuccessful "Gives opinion" I→R interaction increases motivation for a "Disagrees" R→I interaction by a
19. A successful "Gives suggestion" I→R interaction increases motivation for a "Agrees" R→I interaction by a
20. An unsuccessful "Gives suggestion" I→R interaction increases motivation for a "Disagrees" R→I interaction by a

Table 6 Rules for the computation of reactive energy

5.4.2.1 Reactive Motivation from Physical Immediacy

When the agents acting as group members are connected to a representation in 3D space, as will be described in the next chapter, an additional source of reactive energy based on physical immediacy is activated.

According to Latané's (1981) social impact theory, the influence that group members exert on one another is dependant on their physical presence. People positioned in the direct proximity of other group members have a bigger impact on the final group's behaviour. Since, in our model, influence results from interactions, we simply add a given amount of reactive energy to every interaction type and every member perceived by another member in its nearby environment. Contrary to what happens with the previous mechanism, reactive energy is here added every timestep and not only when an interaction finishes.

Given:

- a perceived group member index $m \in \mathbf{Z}_{+}$,
- an arbitrary amount of activation energy $a \in \mathbf{R}_{+}$

$$\textit{for every } i: \quad R_{mi} = R_{mi} + \varpi a$$

The added quantity of reactive energy is then subtracted from the available quantity of proactive energy in the same way as previously described.

This simple mechanism enforces the embodied and environmentally-situated nature of our agents' behaviour, which, according to Dautenhahn (1999), has particularly important properties.

5.4.3 Interaction Triggering

An interaction *mi* is triggered when:

- the interaction initiator is not currently involved in a dyadic or group interaction
- the sum of proactive and reactive activation energy surpasses a given threshold; given a threshold global parameter representing the necessary amount of activation energy for one interaction to become active $\theta \in \mathbf{R}_{+}$

$$P_{mi}+R_{mi}>\theta \quad \mathbf{T}$$

- the sum of proactive and reactive activation energy is greater than the one of all other actions which also surpass the threshold

As will be described in the implementation section, a number of messages are then exchanged by both agents, using a dedicated protocol, in order to decide if the interaction is applicable. Typically, the recipient *m* may not be available. Finally the interaction takes place, or if it is refused, another action whose activation energy surpasses the threshold is selected (or no action is executed at the current timestep).

When an interaction has taken place, the group member's specific interaction drive is satisfied. Two strategies can then be used in order to lower his/her motivation to trigger that same interaction again:

1. **Remove proactive energy**: Since the amount of proactive energy can be negative, this has the effect of maintaining the number of interactions occurring in the group context at a level corresponding to the dyadic rates, the reactive mechanism acting in this case only to order the sequence of interactions.

2. **Remove reactive energy**: Removing the positive amount of reactive energy triggers an adaptation of the dyadic interactions both in their sequencing and occurrence rates, making the close repetition of identical behaviors more probable.

Given:

- a global parameter determining the repartition of energy removal $\psi \in \mathbf{R}$: 0.0 <= ψ <= 1.0,
- the amount of remaining reactive energy $V \in \mathbf{R}$

$$V_{mi}=R_{mi}-\psi\theta$$

Given:

- $V_{mi}>=0$

$$\textit{for every m,i:} \quad P_{mi} = P_{mi} - (1.0 - \psi)\theta$$

$$\textit{for every m,i:} \quad R_{mi} = R_{mi} - \psi\theta$$

Given:

- $V_{mi}<0$

$$\textit{for every m,i:} \quad P_{mi} = P_{mi} - (1.0 - \psi)\theta + L_{mi}$$

$$\textit{for every m,i:} \quad R_{mi} = 0.0$$

Non-dyadic interactions at the group level are not presented because their triggering mechanism is identical, i.e. they are processed as additional competing behaviors. The main difference, in the simulation, is that group interactions involving all group members do not occur as frequently because they require participants not to be already involved in dyadic interactions.

The mechanism we have presented can be easily configured by setting six main parameters which can be described as follows: system heat θ, reactivity ϖ, reactivity decay δ, rates adaptation ψ, simulation timescale τ, random influence ρ. The flexibility provided by these easily adjustable parameters allows for a better control of the resulting simulation.

5.5 A Model of Interpersonal Relationship Development

Small groups are places of intense interpersonal relationship development: people come together, build and maintain intimate relationships but also fall out with each other or simply avoid any contact. As previously discussed, we consider that relationships develop, stabilize or deteriorate through a continuous stream of social interactions. In this section, we present a model where IPA interactions trigger specific micro-changes in the four relationship dimensions we have selected: Liking, Familiarity, Trust, Commitment.

Using the specific properties of IPA interactions, we simulate the evolution of the relationships along the five stages of Knapp's (1996) escalation model, taking into account the characteristics of each stage as they have been documented. A feedback mechanism consecutively modifying interaction rates and average success is also presented.

5.5.1 Liking Development

"Liking" denotes group members' emotional attraction to one another. Rubin's (1973) conceptualisation of liking underlines affection and respect as its two major components, and most definitions include the idea of social "closeness". Interestingly, it has been demonstrated that at lot of energy is required in order to raise and maintain this dimension of a relationship at high levels. For this reason, we use a root function in order to compute individual increases/decreases in liking once an interaction of the Social-emotional type has been completed. The group member's emotional stability is taken into account in order to determine the level of influence of each interaction. Given:

- the interaction success $s \in \mathbf{R}$: $0.0 <= s <= 1.0$,
- a weight relative to the type of interaction (positive for "Shows solidarity" > "Shows tension release" > "Agrees", negative for "Shows antagonism" < "Shows tension" < "Disagrees") $w \in \mathbf{R}$: $-1.0 <= w <= 1.0$,
- a global parameter determining the speed of overall relationship development $\varsigma \in \mathbf{R}$: $0.0 <= \varsigma <= 1.0$,
- a global parameter determining the speed of liking development $\lambda \in \mathbf{R}$: $0.0 <= \lambda <= 1.0$,
- the group member's emotional stability parameter $e \in \mathbf{R}$: $0.0 <= e <= 1.0$,
- the current Liking value $L \in \mathbf{R}$: $0.0 <= L <= 1.0$,
- the new Liking value $L' \in \mathbf{R}$: $0.0 <= L' <= 1.0$

$$L'_m = \sqrt{L_m^{\ 2} + sw\varsigma\lambda(1.0 - e)}$$

5.5.2 Familiarity Development

As group members interact, they get to know each other, which increases their ability to predict the other's behaviour and allows for a better communication and synchronization of their actions. We refer to this important aspect of interpersonal relationship as "Familiarity". The concept does not overlap with "Liking", because it is very common for a relationship to develop on the familiarity dimension without a corresponding growth in emotional attraction (e.g. coworkers), the opposite being also true to a certain extent. We consider that every interaction type contributes to increase familiarity and, in the absence of any well-documented trend, we choose to apply linear growing. Given:

- a global parameter determining the speed of familiarity development $\eta \in \mathbf{R}$: $0.0 <= \eta <= 1.0$,
- the current Familiarity value $F \in \mathbf{R}$: $0.0 <= F <= 1.0$,

- the new Familiarity value $F' \in \mathbf{R}$: 0.0 <= F' <= 1.0

$$F'_m = F_m + \varsigma\eta$$

5.5.3 Trust Development

The definitions of "Trust" in psychology are often very broad, and are based on such concepts as dependability, i.e. the feeling that you can rely on your partners when it matters, faith, i.e. the belief in a relationship continuing indefinitely, etc. This concept has also been extensively discussed in the literature about autonomous social agents by Castelfranchi (2000). In our work, "Trust" is important because high ratings on this relational dimension are required for intimate disclosures to take place. We focus on the informational reliability of the participants, comprised of both objective correctness and subjective intentionality. Typically, an information given with positive intention but which turns out to be wrong and an uncovered lie, will both result in a trust decrease in the receiver (higher when deception is perceived). A low success rating for an interaction of one of the three Task-oriented/Attempts types will produce the same effect in our simulation. We directly link the evolution of trust with the interaction success probability but the effect of an interaction is relative to the current level of trust. Our view, consistent with the theory of expectations, is that what modifies the relationship between one person and another is not any action itself, but the perceived distance between the actions' effects and participants' expectations. In order to match this principle, we use a cubic function that triggers limited changes when an interaction success is close to the current trust value, and high (positive or negative) when the difference is significant. Given:

- a weight relative to the type of interaction ("Gives orientation" > "Gives opinion" > "Gives suggestion") $w \in \mathbf{R}$: 0.0 <= w <= 1.0,
- a global parameter determining the speed of trust development $\varepsilon \in \mathbf{R}$: 0.0 <= ε <= 1.0,
- the current Trust value $T \in \mathbf{R}$: 0.0 <= T <= 1.0,
- the new Trust value $T' \in \mathbf{R}$: 0.0 <= T' <= 1.0

$$T'_m = T_m + 4w\varsigma\varepsilon\left(1.0 - e\right)\left(s - T_m\right)^3$$

5.5.4 Commitment Development

"Commitment" can be described as a social force acting for the survival of the relationship in the future. Studies suggest that commitment does not grow sequentially but that it generally occurs in bursts when enough positive interactions have accumulated, on special occasions, through rituals (e.g. wedding), etc. Therefore, we do not update it when an interaction is completed but only as one of Knapp's relational stages is reached. In order to choose each stage's commitment value, we refer to Knapp's comments which underline that most of the commitment is built in the later stages of a relationship, and decide to use a quadratic function taking as input Liking, Familiarity and Trust values (their role in stage transition will be described later). Given:

- the new Commitment value $C' \in \mathbf{R}$: 0.0 <= C' <= 1.0

$$C'_m = \left(\frac{L_m + F_m + T_m}{3}\right)^2$$

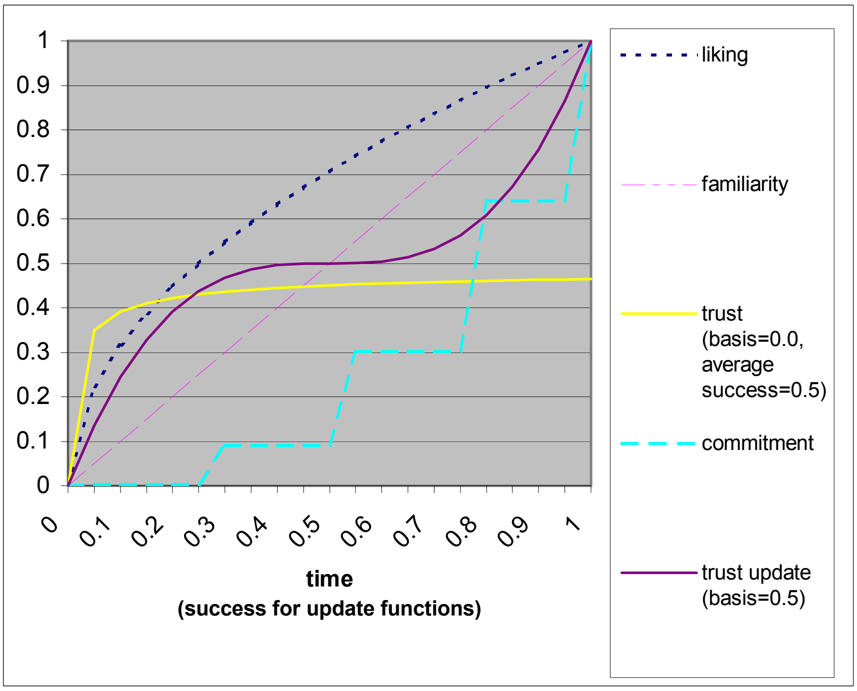


Figure 49 General shape of relationship development functions

5.5.5 Decay Functions

In absence of a continuous stream of interactions, relationships naturally tend to weaken and disappear over time (Burt 2000). By introducing an automatic relationship decay mechanism, we simulate this aspect of relational evolution. Combined with growing functions, the introduction of this mechanism has another important property: it counterbalances the effect of interactions and allows relationships to reach an equilibrium point instead of growing indefinitely when positive interaction rates are higher than negative ones.

Unfortunately, empirical data are missing in order to choose a decay function for our interpersonal relationship dimensions. We have to make a specific assumption: there exists a structural homology between the mental processes leading to relationship decay and the memory mechanisms of forgetting.

Basing our work on this assumption, we reviewed the existing literature describing forgetting curves, and it appears that power functions are generally favoured (Crovitz 1974) in order to model the rapid forgetting following initial retention which can be observed. However, most of these experiments test short-term memory of non-emotional content. Researchers like Linton (1975) or Wagenaar (1988), who have studied more specifically retention of personal episodic material, like interactions involving strong emotions, conclude that forgetting occurs at a nearly linear rate. Therefore, we apply, each timeframe, a linear decay function to update Liking, Familiarity and Trust (we do not consider Commitment decay) values. In order to take into account Commitment's previously described features, we use it to compute the slope of the Liking decay function. Given:

- a global parameter determining the overall decay speed $\iota \in \mathbf{R}$: 0.0 <= ι <= 1.0,
- a global parameter determining the speed of Liking decay $\beta \in \mathbf{R}$: 0.0 <= β <= 1.0,
- a global parameter determining the speed of Familiarity decay $\chi \in \mathbf{R}$: 0.0 <= χ <= 1.0,
- a global parameter determining the speed of Trust decay $\varphi \in \mathbf{R}$: 0.0 <= φ <= 1.0

$$L_m' = L_m - \iota\beta \frac{2C_m + (1.0 - e)}{3}$$

$$F_m' = F_m - \iota\chi$$

$$T_m' = T_m - \iota\varphi(1.0 - e)$$

5.5.6 Relational Stages

In the same way as interactions modify relationships, when the accumulated amount of change reaches a given threshold (Knapp's "relationship turning point"), relationships influence back interaction rates and success. We consider the evolution of interpersonal relationships along the five stages of Knapp’s escalation model, and trigger this feedback mechanism when a movement from one stage to another is detected. Here is a short description of the five stages, of which more details can be found in Knapp's work (1996):

1. **Initiating**: The Initiating stage is a very short stage with a duration typically between 10-15 seconds and a few minutes. It incorporates a series of formal behaviours and conventional formulas (e.g. greetings) that are used when people first come together with other people. Careful observation and initial evaluation of the other takes place.
2. **Experimenting**: The process of Experimenting starts once communication has been established in order to minimize uncertainty. To become acquaintances, the participants must gather a number of information about the other's identity, personality, preferences, etc. Small talk is very important for this purpose and allows for the further evaluation of the person. Many relationships do not develop beyond this stage.
3. **Intensifying**: Careful self-disclosure and more informal behaviours (in the form of address, language, etc.) can be observed during the Intensifying stage. In addition, the accuracy of the predictions about other's behaviours has now improved considerably. As participants consider each other as "friends", some expressions of commitment begin to appear.
4. **Integrating**: Emotional interactions are a lot more frequent in this stage, for instance expressions of empathy, as well as body contacts and other nonverbal behaviours related to a high intimacy relationship. The sharing of ideas and interests becomes very common. Commitment grows as a shared relational identity starts to form.

5. **Bonding**: Bonding is the stage of maximum commitment. The relationship becomes institutionalised, frequently with formal announcement and rituals (also true to some extent for same-sexed pairs). The relationship is now highly personal and emotional. This stage is rarely reached.

The relative duration, the clearing conditions and the effects of each stage we incorporate in our model are inspired from Knapp's descriptions. For one stage to be cleared, we consider that a minimum value must be reached in Liking, Familiarity and Trust (Commitment is indirectly taken into account since it acts as a resistance force to Liking decay), which corresponds to a specific intimacy level. As stages are passed, Commitment increases, the average success and overall number of interactions grow, and interactions tend to become more and more emotional. Table 7 contains a description of each stage transition, given:

- a global parameter determining the amount of feedback on interaction rates $\phi \in \mathbf{R}$: $0.0 <= \phi <= 1.0$,
- a global parameter determining the amount of feedback on interaction average success $\xi \in \mathbf{R}$: $0.0 <= \xi <= 1.0$

Stage	Clearing conditions	Clearing effects
Initiating	-	-
Experimenting	• Current stage is "Initiation". • Liking >= 0.05 • Familiarity >= 0.05 • $\frac{L_m + F_m + T_m}{3}$ >= 0.05	• "Asks for opinion" rate $r = r + 2\phi$ • "Asks for opinion" average success $s = s + 2\xi$ • "Gives opinion" rate $r = r + 2\phi$ • "Gives opinion" average success $s = s + 2\xi$ • All other interactions' rate $r = r + \phi$ • All other interactions' average success $s = s + \xi$ • $C'_m = (\frac{L_m + F_m + T_m}{3})^2$
Intensifying	• Current stage is "Experimenting". • Liking >= 0.25 • Familiarity >= 0.25 • $\frac{L_m + F_m + T_m}{3}$ >= 0.25	• "Agrees" rate $r = r + 2\phi$ • "Agrees" average success $s = s + 2\xi$ • "Gives suggestion" rate $r = r + 2\phi$ • "Gives suggestion" average success $s = s + 2\xi$ • "Asks for suggestion" rate $r = r + 2\phi$ • "Asks for suggestion" average success $s = s + 2\xi$ • "Disagrees" rate $r = r + 2\phi$ • "Disagrees" average success $s = s + 2\xi$ • All other interactions' rate $r = r + \phi$ • All other interactions' average success $s = s + \xi$ • $C'_m = (\frac{L_m + F_m + T_m}{3})^2$
Integrating	• Current stage is "Intensifying". • Liking >= 0.55 • Familiarity >= 0.55 • $\frac{L_m + F_m + T_m}{3}$ >= 0.55	• "Shows tension release" rate $r = r + 2\phi$ • "Shows tension release" average success $s = s + 2\xi$ • "Gives orientation" rate $r = r + 2\phi$ • "Gives orientation" average success $s = s + 2\xi$

		• "Asks for orientation" rate $r = r + 2\phi$ • "Asks for orientation" average success $s = s + 2\xi$ • "Shows tension" rate $r = r + 2\phi$ • "Shows tension" average success $s = s + 2\xi$ • All other interactions' rate $r = r + \phi$ • All other interactions' average success $s = s + \xi$ • $C'_m = \left(\frac{L_m + F_m + T_m}{3}\right)^2$
Bonding	• Current stage is "Integrating". • Liking >= 0.8 • Familiarity >= 0.8 • $\frac{L_m + F_m + T_m}{3}$ >= 0.8	• "Shows solidarity" rate $r = r + 2\phi$ • "Shows solidarity" average success $s = s + 2\xi$ • "Shows antagonism" rate $r = r + 2\phi$ • "Shows antagonism" average success $s = s + 2\xi$ • All other interactions' rate $r = r + \phi$ • All other interactions' average success $s = s + \xi$ • $C'_m = \left(\frac{L_m + F_m + T_m}{3}\right)^2$

Table 7 Relational stages transitions

5.6 Group Development Stages

In addition to the dynamics of interpersonal relationship development, the groups' life-cycle itself has to be taken into account in the simulation.

According to Tuckman (1965), the most frequently observed types of interaction vary over time in most groups. Moreover, the favoured interaction types at any given phase of a group's history appear similar from one group to another. This pattern is related to the response to functional requirements that any group must provide and to what Bales describes as "agenda topics". Tuckman (1965) identifies five stages of small group development which describe the evolution of a collection of individuals to a cohesive group:

1. **Forming**: In this stage, group members are mainly driven by the desire to be accepted by the others and therefore try to avoid any controversy or conflict. Since individual roles and responsibilities are unclear, the main activity is to gather information about other members and about the group's tasks: "Why are we here ?", "What am I/are you supposed to do ?", "How are we going to get it done ?" are typical examples of the intense questioning activity which takes place in the Forming stage.
2. **Storming**: During the Storming stage, tension and conflict arise about group norms and member's status definition. Emotional interactions (predominantly negative) constitute the main group activity, and trigger the formation of interpersonal alliances as well as opposite factions. The group is only marginally concerned with task-oriented behaviours.
3. **Norming**: Group unity forms in the Norming stage, as agreement, consensus and mutual trust grow. It is observed that conflicts cease, interpersonal relationships significantly improve and an acceptable definition of norms and roles is found by a

majority of the group. Through an intense chatting activity, group members share ideas and give or receive positive emotional feedback.

4. **Performing**: Not all groups reach this stage in which members concentrate on the work to be done. They are now interdependent to a significant degree, and the group as a whole is able to achieve more than what the sum of individuals' efforts would. Problems are solved without generating conflicts, and a limited but adequate part of the group's activity is dedicated to the maintenance of positive interpersonal relationships.
5. **Adjourning**: In this phase, the group is disbanded. It may have accomplished its prescribed role or exhausted the available resources.

Although these phases have a number of similarities with Knapp's relational stages, they address a different kind of needs related to the group functioning itself. Both types of development stages are complementary rather than conflicting.

We use group stage transitions in order to apply specific reinforcement to the group member interactions rates, and determine the active stage by computing the average of all interpersonal relationships' Familiarity scores. Contrary to the relational stages, group stages do not work as a purely additive system, i.e. interaction rates are not only increased when a stage is cleared but can also be lowered. We also consider the case where a group moves back to a previous stage, typically because of a change in the group composition. These two factors require that, when a stage transition is detected, the preceding rates update be cancelled before applying the new stage's features. This way of updating the interaction rates provided in the SIML file has the advantage of taking into account the regularities observed on a large number of groups while retaining the specificity of each particular relationship. Table 8 describes the group development stages' activation condition and reinforcement effects[6], given:

- a global parameter determining the amount of feedback on interaction rates resulting from group stage transitions $\phi' \in \mathbf{R}$: $0.0 <= \phi' <= 1.0$

Stage	Required average familiarity F	Clearing effects
Forming	$0.0 <= F < 0.05$	• "Asks for orientation" rate $r = r + 3\phi'$ • "Asks for opinion" rate $r = r + 2\phi'$ • "Asks for suggestion" rate $r = r + 2\phi'$ • "Gives orientation" rate $r = r + \phi'$ • "Gives opinion" rate $r = r + \phi'$ • "Gives suggestion" rate $r = r + \phi'$ • "Shows antagonism" rate $r = r - 2\phi'$ • "Shows tension" rate $r = r - 2\phi'$ • "Disagrees" rate $r = r - \phi'$
Storming	$0.05 <= F < 0.25$	• "Disagrees" rate $r = r + 3\phi'$ • "Shows tension" rate $r = r + 2\phi'$ • "Shows antagonism" rate $r = r + 2\phi'$ • "Gives orientation" rate $r = r + 2\phi'$

[6] The Adjourning stage is not included in our model.

		• "Agrees" rate $r = r + \phi'$ • "Shows solidarity" rate $r = r + \phi'$
Norming	$0.25 <= F < 0.5$	• "Agrees" rate $r = r + 3\phi'$ • "Shows tension release" rate $r = r + 2\phi'$ • "Shows solidarity" rate $r = r + 2\phi'$ • "Gives orientation" rate $r = r + 2\phi'$ • "Gives opinion" rate $r = r + 2\phi'$ • "Gives suggestion" rate $r = r + 2\phi'$ • "Asks for orientation" rate $r = r + \phi'$ • "Asks for opinion" rate $r = r + \phi'$ • "Asks for suggestion" rate $r = r + \phi'$
Performing	$0.5 <= F <= 1.0$	• "Gives orientation" rate $r = r + \phi'$ • "Gives opinion" rate $r = r + \phi'$ • "Gives suggestion" rate $r = r + \phi'$ • "Asks for orientation" rate $r = r + \phi'$ • "Asks for opinion" rate $r = r + \phi'$ • "Asks for suggestion" rate $r = r + \phi'$ • "Shows tension release" rate $r = r + \phi'$ • "Shows solidarity" rate $r = r + \phi'$

Table 8 Group stages transitions

5.7 Implementation

The system we have implemented runs on the Windows 2000/XP PC platform. It has been developed using Microsoft Visual Studio/Visual C++ 6.0 and the MFC library.

5.7.1 Software Architecture

Figure 50 gives an overview of the system's software architecture. Since autonomy and the ability to communicate are essential features in the definition of agency, we have developed a multi-thread architecture where each agent runs independently and is able to send messages to other agents. A multi-thread rather than multi-process architecture limits the overhead (compared to processes, threads save memory and resources, communicate more easily, etc.) while still allowing the asynchronous activity of the group members.

We use the multithreading facilities provided by MFC. A number of MFC UI threads equal to the number of group members is instantiated by 3DSociogramViewer, an application which is described in more details in Section 5.7.3. MFC UI threads have been chosen over MFC WORKER threads since their corresponding classes include useful messaging functionalities, such as a message queue, methods for sending messages to other threads, etc. Windows semaphores are used in order to avoid any conflict when two threads try to simultaneously update the same memory segment, in most cases located in 3DSociogramViewer. All interactions between group member's threads take place via a simple ACL.

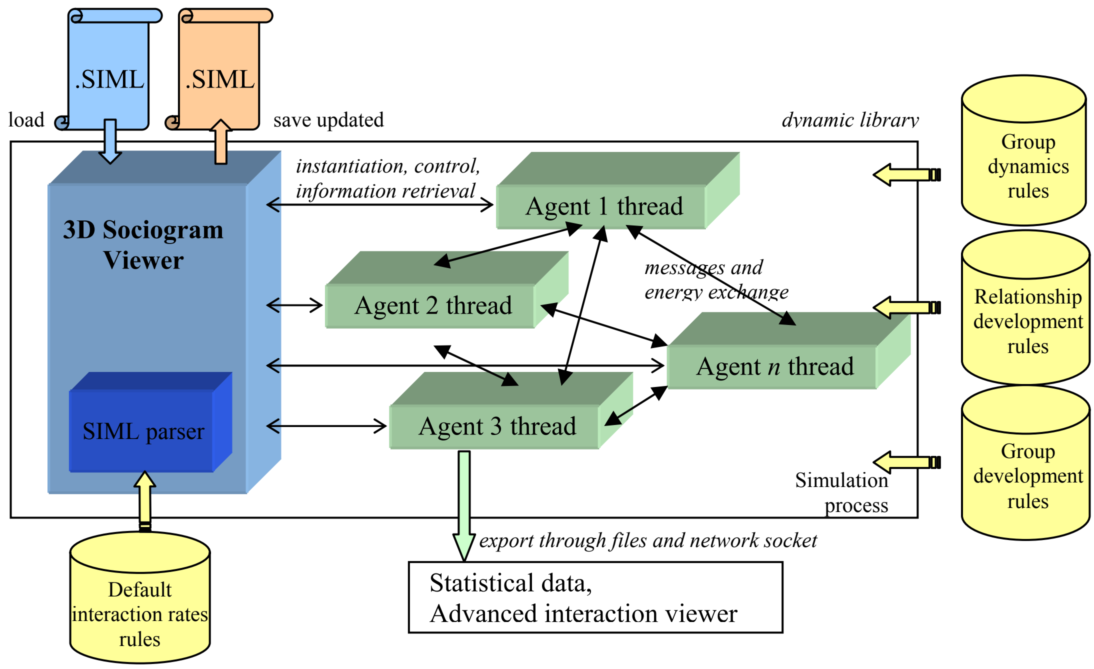


Figure 50 Software architecture of the group dynamics simulation system

The default interaction rates rules, group dynamics rules, relationship development rules, and group development rules are stored in separate dynamic libraries (DLL). This allows a set of rules to be easily replaced by a new one, without requiring the re-compilation of the full system.

5.7.2 Interaction Negotiation Protocol

Using a dedicated protocol, a number of messages are exchanged between threads in order to negotiate the initiation and termination of an interaction. Here is the list of message types:

- M_PROPOSEINTERACTION
- M_ACCEPTINTERACTION
- M_REJECTINTERACTION
- M_TRYLATERINTERACTION
- M_ENDINTERACTION
- M_STOPINTERACTION
- M_PROPOSEGROUPINTERACTION
- M_ACCEPTGROUPINTERACTION
- M_REJECTGROUPINTERACTION
- M_TRYLATERGROUPINTERACTION
- M_ENDGROUPINTERACTION
- M_SETNEXTSUCCESS
- M_ENERGYUPDATE
- M_RELATIONSUPDATE

A number of parameters are passed by reference in the messages. Typically, the interaction type is included in most messages. Additional parameters include: the interaction success calculated by the initiator, an amount of activation energy to be added in the recipient's motivational apparatus, social status information, etc.

A mechanism to avoid deadlock situations (e.g. Agent 1 is trying to interact with 2, which is trying to interact with 3, which is trying to interact with 1) using M_TRYLATERINTERACTION messages has been developed. When an agent is not currently interacting but is already involved in an interaction negotiation, it will ask other requesting agents to resend their interaction proposal at a later time. Each negotiation has a time-to-live value which is decreased every time messages are exchanged. When it has reached zero, the interaction proposal is considered refused and another agent is requested (or if the motivation energy is insufficient, no further action takes places except the handling of other's requests). Complex interlacing of interaction requests involving both dyadic and group interactions can occur. In these cases, it happens that an agent is not able to participate in an interaction when an optimised strategy would have allowed it, however the system never generates infinite loops and blocked threads.

The main low-level rules guiding the interaction negotiating are straightforward:

- When an agent whose motivation threshold for interacting with another agent is reached, it sends a M_PROPOSEINTERACTION message to the corresponding thread.
- When an agent whose motivation threshold for interacting with the group is reached, it sends a M_PROPOSEGROUPINTERACTION message to every thread.
- When an agent is not involved in an interaction and receives a M_PROPOSEINTERACTION message interaction, it replies with a M_ACCEPTINTERACTION message.
- When an agent is not involved in an interaction and receives a M_PROPOSEGROUPINTERACTION message interaction, it replies with a M_ACCEPTGROUPINTERACTION message (a group interaction's success is proportional to the number of positive and negative replies).
- When an agent is involved in an interaction and receives a M_PROPOSEINTERACTION message interaction, it replies with a M_REJECTINTERACTION message.
- When an agent is involved in an interaction and receives a M_PROPOSEGROUPINTERACTION message interaction, it replies with a M_REJECTGROUPINTERACTION message.
- When an agent is involved in an interaction negotiation and receives a M_PROPOSEINTERACTION message interaction, it replies with a M_TRYLATERINTERACTION message.
- When an agent is involved in an interaction negotiation and receives a M_PROPOSEGROUPINTERACTION message interaction, it replies with a M_TRYLATERGROUPINTERACTION message.
- When a dyadic interaction finishes normally (decided by the initiator), the initiator sends a M_ENDINTERACTION message to the other agent's thread.
- When a group interaction finishes normally (decided by the initiator), the initiator sends a M_ENDGROUPINTERACTION message to every involved agent's thread.
- When a problem is detected by the initiator, it sends a M_STOPINTERACTION to the other agent's thread (and the interaction success is set to zero).

High-level rules can be used to enrich the mechanism. We consider that it is not possible, given the social constraints and because it would affect the entire system, for an agent to refuse to interact with another agent when it is not already involved in an interaction. However, we choose to allow an agent to interrupt an ongoing interaction when it receives a proposal from an higher status member:

- When an agent chooses to interrupt an interaction, it sends a M_STOPINTERACTION to the other agent's thread (and the interaction success is drastically lowered).

The probability for an agent to interrupt an interaction is low and proportional to the difference of status between both requesting agents. Given:

- the requesting group member index $m \in \mathbf{Z}_{+}$,
- the interacting group member index $m' \in \mathbf{Z}_{+}$,
- the probability of interruption of the interaction $p \in \mathbf{R}$: $0.0 <= p <= 1.0$

$$p = 0.2 * \left(\frac{Leadership_m + Power_m}{2} - \frac{Leadership_{m'} + Power_{m'}}{2}\right)$$

When the requesting group member's social status is lower than the current interlocutor's, the probability is null in any case.

The basic fact that an agent cannot interact at the same time with several agents except in the special context of group interaction, has a number of consequences which are discussed in the section presenting the properties of the system as a whole.

5.7.3 3DSociogramViewer

The central component in our system is a GUI application called "3DSociogramViewer" (see Figure 51). In addition to MFC, it makes use of the OpenGL graphic library (v1.2) and of the GLUT utility toolkit. The application is in charge of several important tasks:

- loading, merging and saving of SIML files: done through the SIML parser library,
- starting, pausing, accelerating or slowing down of the simulation: 0x, 1x, 10x, 100x the standard timescale,
- display of group's and group members' characteristics (see Figure 52), as well as various statistical indicators:
  - Group cohesion, which is calculated as the average of all liking scores.
  - Power/prestige spread, which is an indicator of inequality: the range and standard deviation is provided.
  - Group performance, whose index is calculated on the basis of the average of every interaction success scores, but also taking into account the group's traits that psychological studies have shown to be relevant: heterogeneous groups in terms of gender and culture perform better than very homogenous ones, performance improves when the leader also has a high score in power, small

groups generally outperform large groups (Ringelmann effect), very low and very high scores in group cohesion have a negative impact on performance (Steiner 1972).

- graphical representation of the group members, interpersonal relationships and interactions (described in the next section),

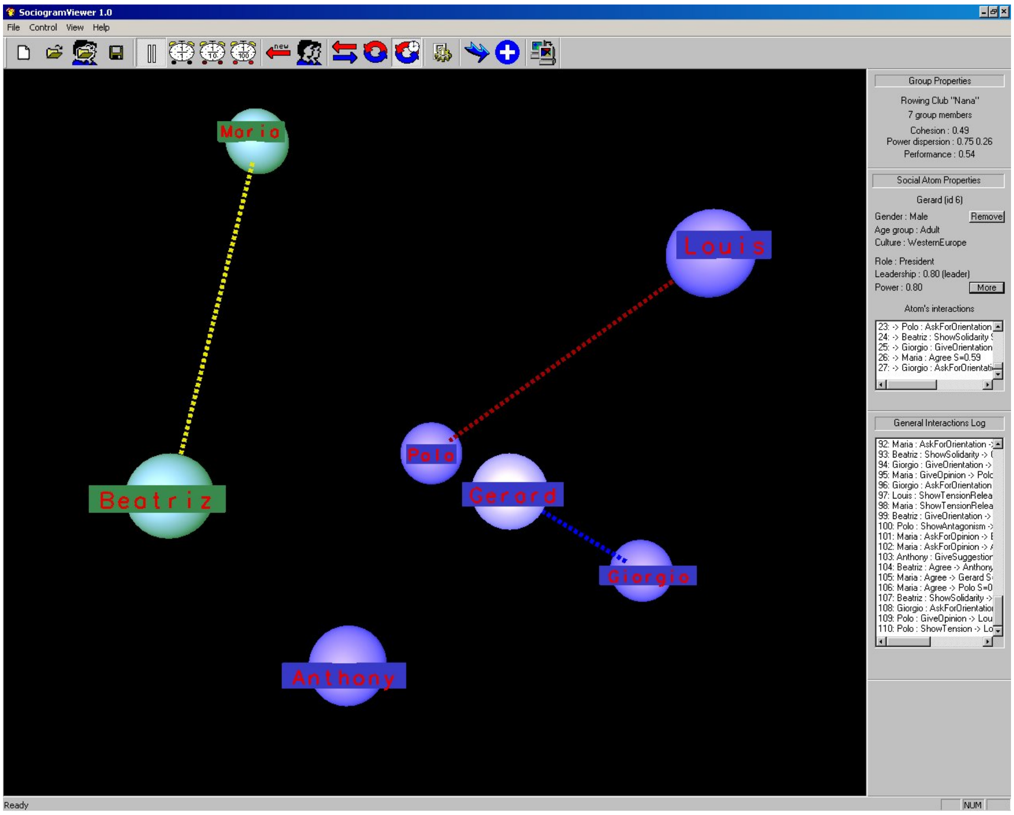


Figure 51 A small group displayed in 3DSociogramViewer

- creation and modification of groups through addition and removal of members (see Figure 53),
- recording and display of past interactions as well as the possibility for the user to generate new ones (see Figure 54),
- setup of the multi-thread architecture,
- generation of various logfiles: interactions log, relationships evolution log, group's statistical indicators evolution log,
- export on a network socket of the interaction data allowing other applications to extend the simulation: group members' characteristics and interactions (participants, interaction type, interaction success),
- parametrization of the group dynamics engine (see Figure 55)

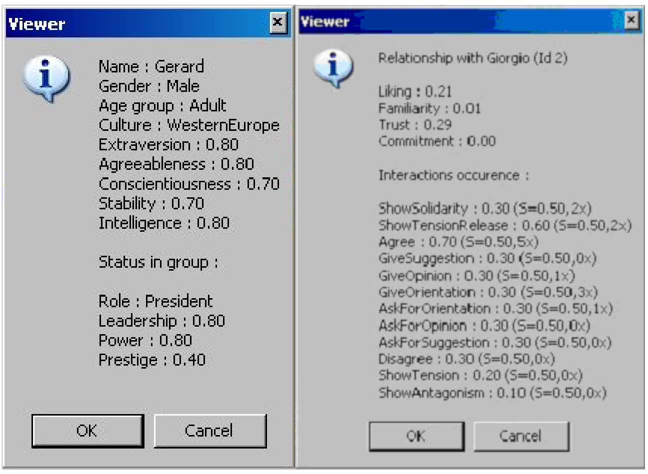


Figure 52 Display of information about a group member

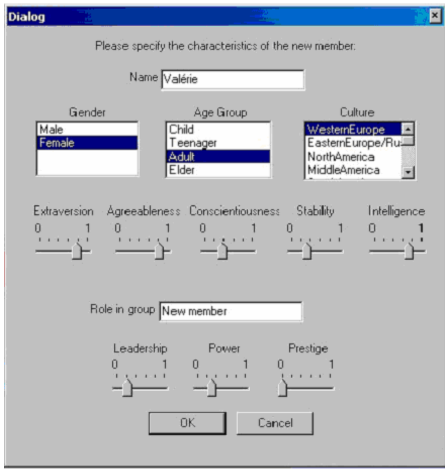


Figure 53 Creation of a new group member

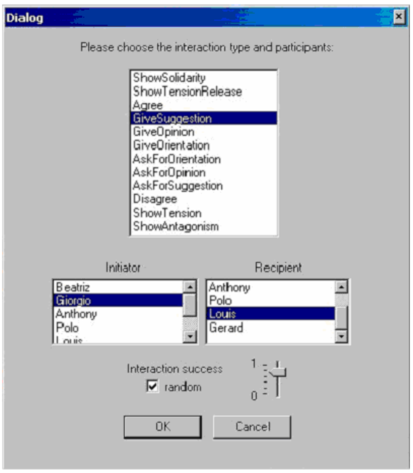


Figure 54 Manual generation of an interaction

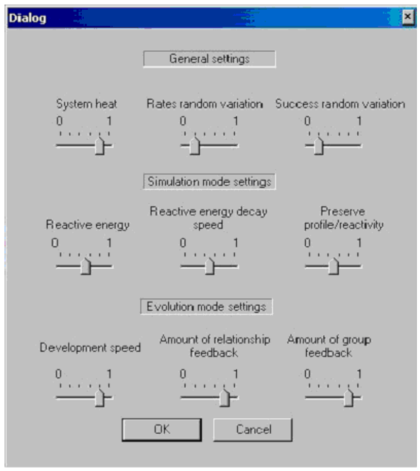
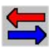


Figure 55 Configuration of the group dynamics engine

5.7.3.1 Operating Modes

The higher-level behavioural layers of our architecture can be activated and deactivated at will. Depending on the application requirements, 3DSociogramViewer can be used in three different ways:

1.  **Control mode**: In the control mode, the interaction rates provided in the SIML file are strictly respected since reactive activation energy is not used. Except for the influence of the random variation parameters, which can be defined in the settings panel, interactions are triggered on a regular basis and can be safely anticipated. This mode is useful when it is important for simulation managers to be able to finely control the group's behavioural outcome. It is also well adapted to SIML data which has been gathered through the observation of real groups and which is not based on calculated dyadic interaction rates.

2.  **Simulation mode**: In this mode, the group dynamics rules are activated in order to produce realistic chains of interactions. This means that, depending on the chosen settings, limited variations in the actual interaction rates compared to the provided interactions rates may be observed. The simulation mode is used when the ordering of interactions is important, for example when they are to be visually simulated, but that the group profile and interaction rates must be preserved.

3.  **Evolution mode:** The evolution mode is used to simulate group behaviours on a long time frame. Every mechanism described in this chapter is activated, including the interpersonal relationship and group life-cycle reinforcements. Because of the complex interactions between these mechanisms, predictions are a lot more difficult to make than in the other modes. On the long-term, dyadic interaction rates can be dramatically transformed when the evolution mode is used.

The active mode can be changed at any time with the 3DSociogramViewer toolbar (see Figure 51).

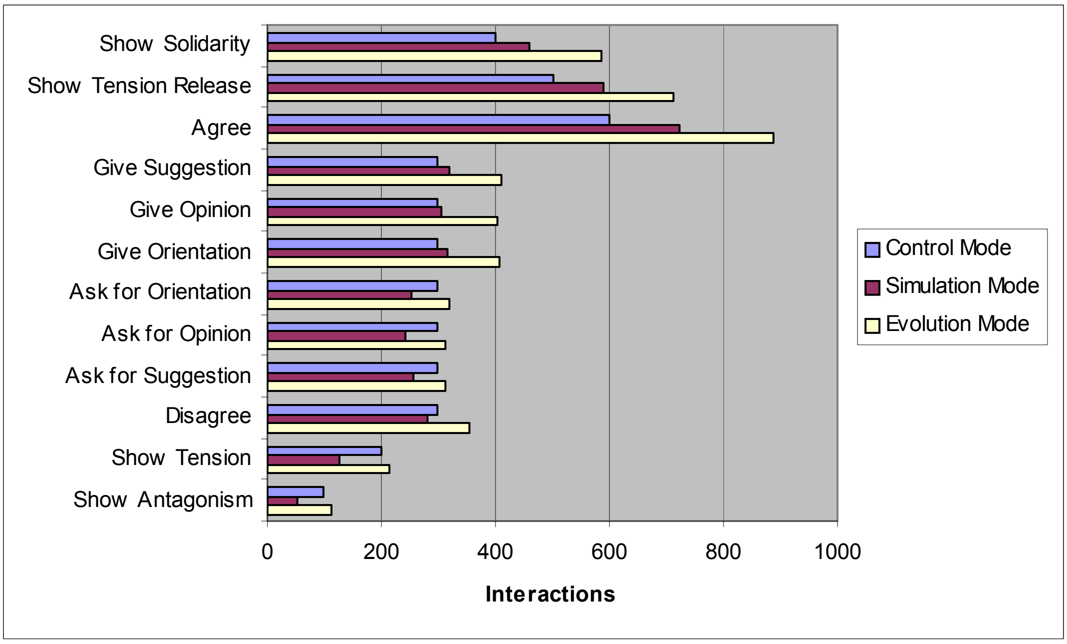


Figure 56 Comparison of the generated interactions in different operating modes

Figure 56 shows a comparison of the overall number of generated interactions in a positive social-emotional group using the three operating modes; when a group must be analyzed in details, similar charts can be created for every dyad. In the Simulation mode, reinforcement of the social-emotional positive interactions can be observed, as well as some inhibition of the negative and task-oriented interactions ("Give Suggestion", "Give Opinion" and "Give Orientation" rates are not lower because they are reinforced by "Ask for Suggestion", "Ask for Opinion" and "Ask for Orientation" interactions). The main effect of the use of the Evolution mode is here a general increase in the interaction rates.

5.7.3.2 Graphical Representation of Small Groups

The graphical representation of experimental data is extremely important in most scientific fields because it is a powerful device for uncovering structural process (Freeman 2000). New ways of representing and arranging the objects of study stimulate the researchers' imagination and intuition, which can generate major advances in the state of the art. In consequences, one of our goals is to provide a powerful and convenient tool for the visualization of small groups and small groups dynamics.

In our work, the graphical representation of group and group members is realized through the display of so-called "sociograms". According to Moreno (1960), social systems are attraction-repulsion systems, and what he calls "sociogram" is a graphic representation of the patterns of inter-members attraction in a group. In sociograms, a circle, or social atom, represents a group member. Social atoms are placed close to each other if their associated individuals are strongly attracted to each other, the preferred group members are placed in the centre of the graph, and lines are drawn for each relationship. 2D graphs following this principle are commonly used to portray various social networks. Some 3D representations based on chemistry visualization tools or on VRML are beginning to appear (Freeman 2000), but, as static views, they do not allow researchers to properly take into account the dynamic nature of social processes.

In order to represent our groups and group members, we generate computer graphics sociograms which have the following novel properties:

- The atoms are positioned and rendered in 3D space, which contributes to a clearer representation than standard 2D graphs, since more room is available.
- Our sociograms can be interactively moved, scaled, rotated, or put in an automatic rotation mode to avoid any visual occlusion.
- By default, only the atom colour is used to indicate gender (blue=male, green=female), but an enriched display mode (see Figure 57) can be activated to provide additional visual clues (metaphorically, the bigger the atom, the higher the ratings in power and prestige; a dodecahedron is used instead of a sphere to represent leaders).
- When an interaction is triggered, an animated stripped line specific to the interaction type is drawn between the initiator and the receiver (thus, lines do not represent relationships like in Moreno's sociograms).
- In our dynamic sociograms, changes in group members' relationships are immediately reflected by atom displacements, which creates a continuous evolution of the sociograms' shape; for example, the force corresponding to Liking decay constantly pulls apart the atoms while positive interactions tend to bring them together.

3DSociogramViewer makes use of a 3D spring embedder algorithm and associated techniques (Simon, Steinbrückner & Lewerentz 2000) in order to position each social atom in space relatively to the others, and computes physical forces in order to represent social forces. More specifically, Liking between participants determines the desirable geometric distance between their corresponding social atoms; we consider that every atom is connected to every other by a spring of length equal to this distance. When the distance between two atoms is greater than the desired distance, the spring is pulled; when it is lower, the spring is pushed;

otherwise, the spring is in a state of equilibrium. Through a number of iterations changing the position of the atoms, starting with randomly chosen coordinates, the spring embedder algorithm tries to minimize the total energy that is given by the summation of these spring forces.

In order to determine how to move one atom in one of the three dimensions so that the new configuration decreases the energy of the whole system, a number of partial derivatives are calculated. We do not describe here the complex procedure used to calculate these derivatives since it is not directly related to our topic, however all the details required for a complete implementation can be found in a paper by Simon, Steinbrückner and Lewerentz (2000).

Generally speaking, the more numerous the group members are, the more difficult it is to find an optimal solution, since an atom has many (contradictory) desired distances and a move might lead to a configuration better representing one relationship and not another. For small groups (up to 20 members), a suitable solution is found with a limited number of iterations, which means that we can continuously apply the algorithm in order to reflect relationship changes.

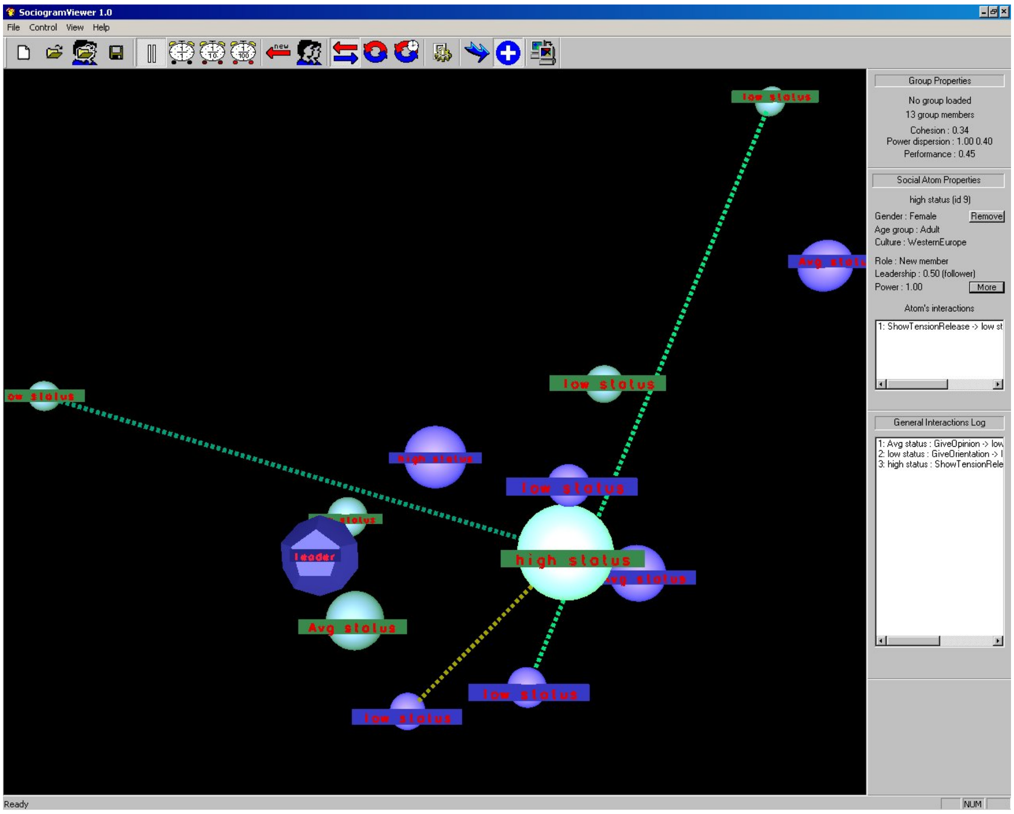


Figure 57 The 3DSociogramViewer enriched display mode

A number of interesting properties result from the integration of the algorithm in the visualization tool. For example, the most "popular" members are automatically placed near the centre of the graph, which corresponds to Moreno's recommendations, without any explicit programming. Another property is best described with the following scenario: when a member positioned near the centre of the sociogram faces the introduction of a new member with even higher attraction ratings (except with the "rival"), the first member appears to be slowly rejected in the periphery of the graph, in spite of his unchanged relationships with original members. This example shows that important features of group life can be underlined using this representation technique.

5.8 Properties of the Models

The integration of the relationship and group development models with the presented group dynamics mechanism, summarized in Figure 58 creates a complex system with emergent properties. When the delayed feedback mechanism is triggered, not only is the relationship between both interactants reinforced but the event also alters the social context and influences all future interactions in the group. The use of a delayed feedback mechanism generating a "threshold effect" instead of continuous reinforcement, is more consistent with empirical observations and does not automatically drive the system to some extreme states, which is the case for some simulation models. An example of interpersonal relationship development in the context of small groups is given in Figure 59.

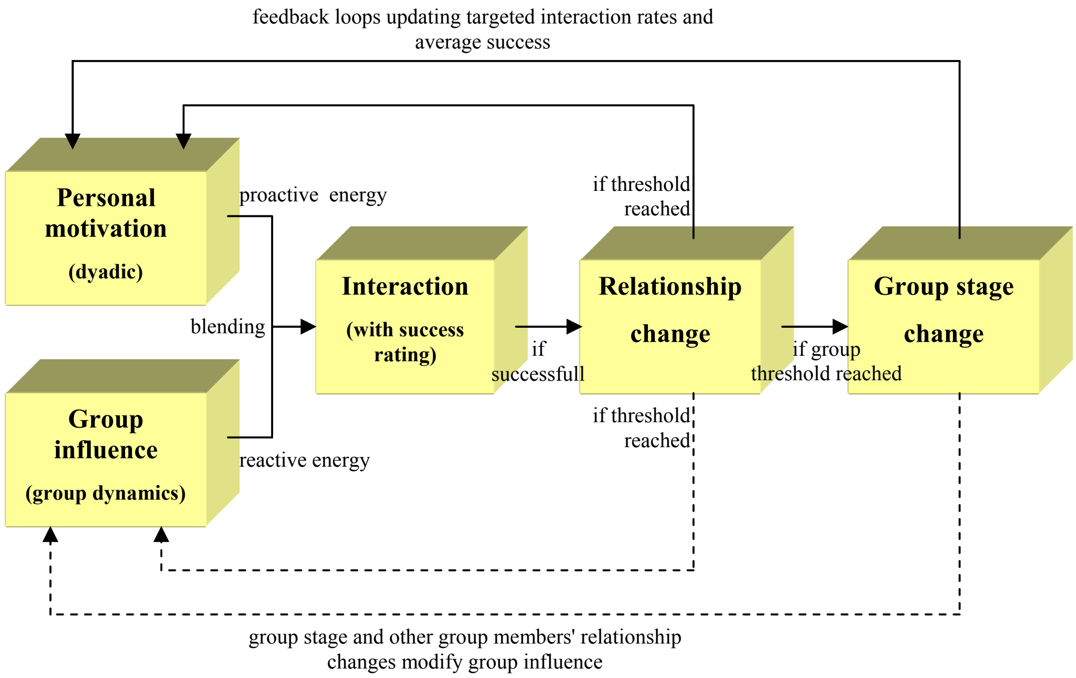


Figure 58 Interaction of the different models

Our investigations of the model reveal a number of important properties which are consistent with existing empirical data (Bales 1951; Hare 1966; Bales 1970). Some properties are directly linked to our design choices but others emerge from the complex interactions of the different model elements. Here are the properties we have identified:

- A process of mutual adaptation between the group and group members can be observed. For example, when a new member with a strong preference for emotional interactions joins a task-oriented group, his/her interaction profile is transformed, as are, to a lesser extent, the other group member's actual interaction rates. The drives of a group member can be directly inhibited by the behaviour of others but it also happens that he/she is so busy responding to other's solicitations that the opportunities to act in a way corresponding to these drives are few. The addition of new group members with similar characteristics gradually allows the expression of these drives, and can even lead to the reverse situation. The more extravert/expressive a group member is, the more influence he/she has on this process. These observations are relevant to social norms theories as well as to minority group studies (Latané 1981).

- Each group will tend to organize chains of interactions in a specific way depending on the characteristics of its members, but a number of regularities can be observed. These regularities are the intended effect of our definition of the interaction rules, which is aimed at reproducing the features described by Bales. The alternating emphasis on social-emotional and task-oriented behaviour is the most important aspect. Although any type of activity may be found occurring at any time, the distribution of specific phases in the time dimension, where a number of interactions of the same type are grouped, clearly appears in the produced behaviour. Interestingly, "Agrees" and "Disagrees" interactions are typical pathways from task-oriented to emotional actions, while "Shows tension release" interactions frequently move the emphasis towards positive emotional interactions when it first was towards negative ones. Conversely, the transition from emotional to task-oriented interactions can only occur when the drive to turn back to the task area has significantly increased. The tendency toward repetition of the received interaction and the organization of question-answer sequences are also confirmed.

- Relationships evolve through a number of semi-equilibrium states (plateaus). Oscillatory fluctuations in Liking, due to the conjunction of automatic decay and interactions influence, can be observed (Figure 59). Through these fluctuations and because increases in Liking are computed by a root function, the system converges and stabilizes around a specific level which is dependent on the starting conditions (interaction rates, group profile and behavioural engines settings, e.g. decay speed). A stronger random variation or an external event (e.g. a member leaves the group), apparently of little importance, can then disturb it just enough for a new threshold to be reached. The resulting increases in rates, interaction success and commitment, modify the level of stabilization, and the resistance to the next stage is challenged. As Knapp (1996) underlines, there is always a certain degree of instability associated with any stable relationship.

- Very few relationships reach the highest development stage. Conversely, almost all relationships reach the Experimenting stage very quickly and the majority stabilizes somewhere in the Intensifying stage. Generally, when a transition threshold is reached through an exceptional random variation or unique event as opposed to a continuous growth or structural event (e.g. a new member joins the group), the next stage will never be cleared.

- Several factors contribute to positive relationship development. In our simulations, similar group members tend to develop better relationships than others whatever the context. Participants with high rates in "Shows tension release" are very efficient at stopping tension escalating loops, which allows relationships to grow consistently. An increase in commitment always raises the average Liking value after some iterations. Group members who interact a lot experience faster relationship evolution, and are generally more popular than others.

- Like in existing groups, when a popular individual is overwhelmed by requests and is not able to participate in a number of positive interactions, the relationship growing rate cannot be sustained, and Liking can even stabilize at a significantly lower level because of the automatic decay function. When all group members have very high interaction rates, temporary interlocking occurs, which causes random interaction selection and leads to chaotic relationship development. Dyads with identical positive and negative interaction rates generally experience stalling of relationship development or slow decay, because positive and negative interactions cancel each other out. This leaves Liking decay as the main determinant, but its influence is somewhat limited by the structure of interaction rules which slightly favours positive evolution.

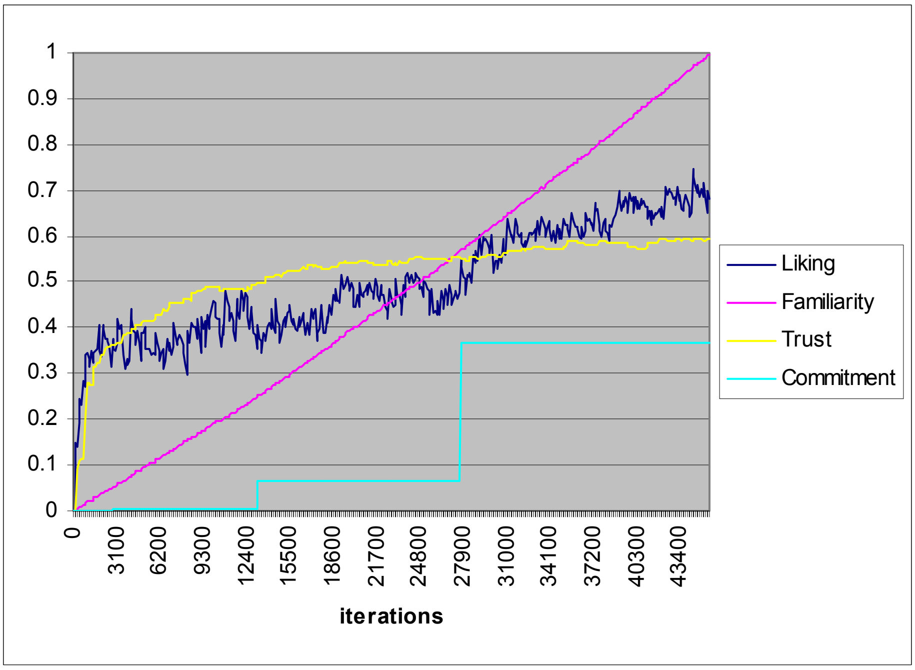


Figure 59 Example of interpersonal relationship development

- The order of interaction triggering is relevant to relationship development. A long series of positive interactions has a greater influence on relationships than mixed sequences, since it facilitates the clearing of a relational stage. The resulting commitment increase makes the relationships more resilient in the sense that once a relationship has reached a given level, it will be able to come back to it a lot faster than the first time, even though a number of grouped negative interactions, corresponding to the series of positive ones, has been triggered.
- Relationships and actual interaction rates tend to become increasingly symmetric in dyads through mutual influence, even for participants with very different profiles at the beginning of the simulation. When the reciprocity factor is high, strong behaviour reinforcement occurs and the development of the relationship is accelerated.
- A group is in a state of full equilibrium only when all its members have separately attained equilibrium, i.e. every relationship has stabilized around a value whose distance from the next stage threshold is higher than the highest possible random variation. As long as at least one relationship continues to evolve, a new stage can still be cleared, and the corresponding update in interaction rates and success may trigger a cascade of new stage transitions. In other words, a disturbance which affects the equilibrium of one member will also affect the others. A stage transition in the group development process and a change in the group personnel are the other main factors which can upset the global equilibrium. When a new (unknown) member joins the group, it automatically lowers the group cohesion, which will later stabilize at a level

depending on the new member's characteristics. The loss of a member or the substitution of one member for another are other typical events which disturb the group equilibrium on the long term. The study of small groups using a systemic approach has generated similar findings (Hare 1966).

- Relationships exposed to different social contexts do not develop in the same way. In our small groups, participants influence each other and their interactions are conditioned by the group profile. Since relationship development occurs through these interactions, the development of their relationships can be greatly modified. What is commonly referred to as "group atmosphere" acts independently of mutual attraction and dyadic interaction rates. Typically, groups with high rates in social-emotional/positive interactions tend to promote successful interpersonal relationship development through fast increases in Liking, while those performing many social-emotional/negative interactions inhibit the process. A strong emphasis on task-oriented interactions slows down the development of the relationship emotional dimension, which can contribute to establish it at a slightly lower level due to automatic decay, but speeds up Familiarity and Trust evolution.

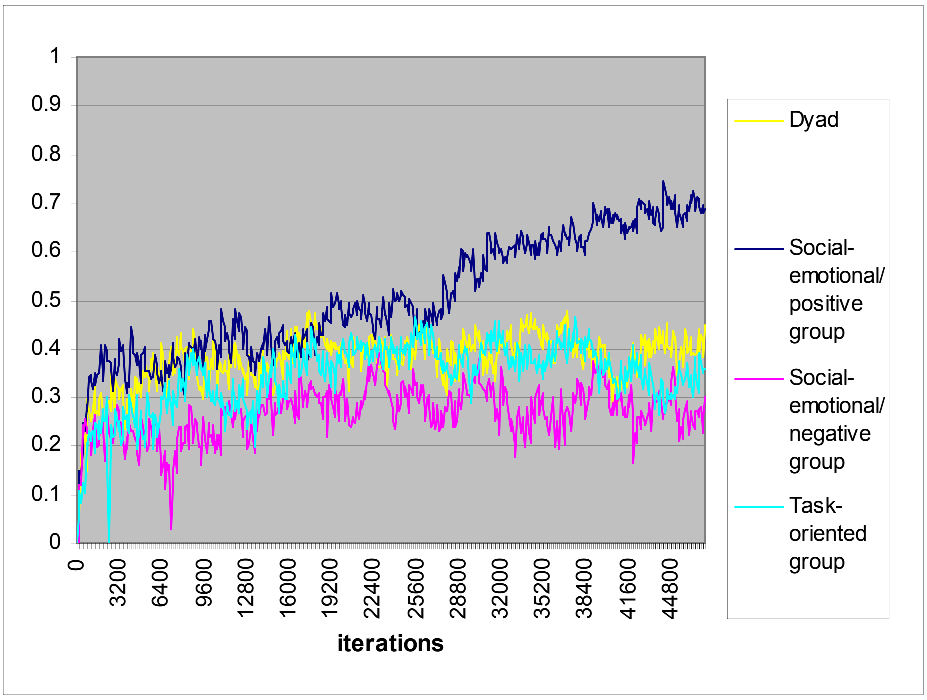


Figure 60 Comparison of Liking development in different group contexts

Figure 60 shows the results of having simulated the development of Liking between two group members of a rowing club. The group profile, i.e. other group members interaction rates, was altered for each simulation, but the rates for the interactions of both subjects were strictly identical each time[7]. Results clearly illustrate some of the described trends, like the influence of the group context on interpersonal relationship development. The social-emotional/positive group settings allowed the relationship to clear the Intensifying stage and to stabilize in the Integrating stage.

[7] Medium to positive interaction profile: Shows solidarity rate=0.4, Shows tension release rate=0.5, Agrees rate=0.6, Gives suggestion rate=0.3, Gives opinion rate=0.3, Gives orientation rate=0.3, Asks for orientation rate=0.3, Asks for opinion rate=0.3, Asks for suggestion rate=0.3, Disagrees rate=0.3, Shows tension rate=0.2, Shows antagonism=0.1.

5.9 Conclusion

The work we have just presented demonstrates how the simulation of what is commonly perceived as a purely psychological phenomenon - the development of interpersonal relationships - can be enriched using a heuristic conceptualisation of sociality. Turning back to Maturana's view (1980) given in the introduction, it appears that our model incorporates the main features of a social system emerging from the interactivity of participants. As noted in the introduction, our groups are more than the simple addition of their dyadic relationships, and, when using the Simulation or Evolution mode, a difference appears between the SIML specification provided in terms of interaction rates and the observed output in terms of interaction occurrences.

The adequate definition of simple rules as well as the intertwining of different forces (psychosocial attraction and interactional logic) and cycles (relationship and group development), proved to be sufficient to generate rich social behaviours. We think that the resulting model exhibits several valuable properties and has a fair amount of explanatory power. However, the degree of realism of the produced behaviours could certainly be increased by better taking into account the environment of the groups and their external constraints. A more in-depth use of the SIML information would also further enrich the simulation.

Bales's IPA categories offer a convenient way of integrating our group dynamics model with the work presented in Chapter 4. IPA interactions could be used to distribute, inside a given group, the data structures proposed for the modelling of social agents: "Gives suggestion" interactions can be used by group members in order to communicate social norms to each others, "Gives opinion" interactions can serve to exchange beliefs (typically about other agents) and "Gives orientation" interactions can be used as a means to alter others' social values. The influence of each group member in this process, calculated for example to determine if a particular member has to accept a major change in its beliefs or can deny it, could be modelled using the SIML power and prestige ratings. The resulting degree of uniformity among group members would thus be dependant on the group's status structure.

In addition to the model itself, we hope that SIML and 3DSociogramViewer can prove useful for other researchers. During the development and testing of the tool, it has been particularly fascinating and stimulating to observe social atoms representing real human beings, pulled apart under the constant influence of Liking decay, having to endlessly fight entropy by generating positive social-emotional interactions. This example illustrates the amount of energy required in order to maintain what sociologists call the Social Link, and is typical of what a rich and dynamic representation of social processes has to offer.

Turning back to the issue of simulating nonverbal social interaction in virtual environments, it now appears that a fundamental requirement is addressed: the system we have described in this chapter provides a continuous stream of face-to-face interactions, which have well known characteristics and involve well known participants. Our next task is to properly visualize these interactions in virtual environments, by generating appropriate nonverbal behaviours composed of face and body animation.

Chapter 6 • Visualization of Social Interactions in Small Groups

6.1 Introduction

In the first part of our research, we identified a number of requirements for the simulation of nonverbal interactions in virtual environments. Chapter 3 presented a series of low-level components for their visualization, and, in the preceding section, we introduced a group dynamics engine which is in charge of the generation and organization in the temporal dimension of a number of well defined interactions. In this final part of our research, we now bring together these contributions to propose a solution for the 3D simulation of nonverbal social interactions taking place in the context of artificial small groups.

Our simulation is focused on the two main dimensions of interpersonal nonverbal behaviours:

- **Proxemics**: Proxemics refers to the way participants use the space around them. This dimension includes interpersonal distances and interaction angles, which Chapter 2 has shown must be addressed, but also the behaviours associated with the collision avoidance problem. To simulate the proxemic aspects, our work is based on Hall's (1966) personal space theory and on Ciolek and Kendon's (1980) taxonomy of possible formations in face-to-face encounters.
- **Kinesics**: Kinesics is related to face and body movements and includes the types of action used in the experimental nonverbal communication interface: among others, gestures, facial expressions and postural shifts. We use Ekman and Friesen's (1969) classification of nonverbal communication signals in order to take into account the specificity of every type of action. In addition, Mehrabian's (1972) research on postures and Argyle and Dean’s (1965) Equilibrium Theory have proved extremely valuable in our work.

In order to simulate the behaviours related to the proxemic and kinesic aspects, the main parameters influencing nonverbal communication mechanisms must be identified. According to Corraze (1980), three different types of factors affect these mechanisms. Nonverbal acts can be simultaneously:

- the expression of a particular social identity,
- the expression of a particular interpersonal relationship,
- the expression of an emotional state or of an instrumental requirement

The work presented in the preceding chapter allows for a rich social characterization of face-to-face encounters which includes these three dimensions. The SIML data attached to an interaction provides a large amount of information about the participants' social identity, their relationships and the interaction itself.

Consequently, the next section of this chapter is dedicated to the establishment of a link, using an adequate theoretical framework, between these parameters and the nonverbal behaviours to generate. The third section describes every step involved in the management of nonverbal interactions in our system, for dyadic as well as group interactions. In the fourth section, we present three parametric animations we have developed on the basis of the selected theoretical framework. It will be explained how interaction postures, observation behaviours, and a series of actions that ensure the continuous nonverbal activity of the participants (a requirement identified in Chapter 2) are generated. Our choices for the simulation of the proxemic aspects of nonverbal communication, including the social avoidance of collisions, are explained in Section 6.5. In Section 6.6, we introduce the Social and Emotional Displays Markup Language (SEDML) and present an approach, based on Argyle and Dean’s Equilibrium Theory, for the generation of kinesic behaviours. Finally, we discuss the integration of our work in the VHD++ framework (Ponder et al. 2003) and with the group dynamics simulation system presented in Chapter 5, and conclude with an example of the produced behaviours in a complex 3D environment.

6.2 Theoretical Framework

According to Ketrow (1999), research examining nonverbal behaviours in groups can be divided into two main categories: (1) the context-specific study of groups in organisations like companies, military, or schools, (2) the study of nonverbal activity observed in "social groups", such as groups in pubs, sport clubs, support/self-help groups, "street-corners" groups, etc. The first body of research is oriented to task groups and typically makes use of variables linked to the hierarchical status in order to explain behavioural variations. In the tradition of social groups study, the findings based on the observation of dyads are generalized to the group context. This approach appears better suited to our needs.

The nonverbal behaviour research problem has been tackled from different angles by several authors. Some researchers have proposed classifications of nonverbal acts according to the communication channel used, others have described the functional requirements they fulfil (Ekman and Friesen 1969; Corraze 1980; Patterson 1983; Burgoon, Buller & Woodall 1996). However, as it has been said in the introduction, what is needed for our research is the uncovering of the connection between social-psychological characteristics of our group members and interactions (represented with SIML data) and a given behavioural output.

When reviewing the psychological literature in search of the main determinants of nonverbal behaviour, it appears that numerous contributions have been made. However, most studies are focused on the impact of one specific variable on one specific aspect of nonverbal activity. Here are some examples from Hayduk's (1983) compilation of 25 studies identifying relevant variables:

- **Gender**: Males use larger interpersonal spaces than females, women are more skilful than men at decoding nonverbal cues.
- **Age**: Interpersonal distances increase regularly between the ages of 2 and 20.
- **Culture**: Israeli subjects stand closer to each other than Swedish subjects, but choose positions allowing them to keep an eye on the surroundings, rather than on each other.
- **Personality**: Introverts and extroverts differ significantly in their resistance to spatial violations, preference for erect posture correlates with intelligence.

- **Social status**: High status individuals establish a greater interpersonal distance than low status individuals and are more successful in keeping their territory free of intrusion, persons holding power positions approach others more closely.
- **Relationship**: Friends tolerate closer presence.

Other aspects which have been studied in details include the influence of the environment like temperature or seats arrangements, the participants' stress level, the type of task to perform, etc. It appears that nonverbal behaviours are subject to a great deal of variation and that a virtually endless list of factors can be identified. Using this research strategy, the human populations can be divided and subdivided endlessly, and it is therefore difficult to integrate consistently these findings in a simulation system. A possible approach would consist in preparing a number of plans (e.g. AML plans) that describe the nonverbal behaviour of, for example, a female agent with a male agent when the interaction is of the "Shows solidarity" type, an adult agent in a "Shows tension" interaction with a child, etc. Such a procedure may give good results but is extremely tedious, and it is very probable that not all possible combinations can be provided by simulation designers. Therefore, we choose a different method: when it is applicable, we want to use a limited number of high-level parameters, calculated on the basis of SIML data and IPA interaction types, in order to drive the participant's nonverbal behaviour.

What is needed is an unified theory of nonverbal behaviour, and, in this context, Mehrabian's (1972) three-dimensional framework is particularly remarkable. Its main advantage is that it allows for the complete but succinct characterization of nonverbal behaviours. Most nonverbal signals can be adequately described in this framework, only some purely regulative behaviours used for the management of the interaction (e.g. to signal turn-taking or turn continuation, as back-channel reinforcement, etc.) do not appear to exist in one or another model's dimensions. Mehrabian's three dimensions of nonverbal behaviour are Immediacy, Relaxation and Responsiveness:

1. **Immediacy**: Immediacy is defined as ”the perceptual availability of persons to one another” (Mehrabian & Diamond 1971, p. 282). When high immediacy interactions take place, the mutual sensory stimulation is increased between the participants: characteristically, they touch each other on a regular basis, are positioned close together, lean forward toward the addressee, maintain eye contact, and adopt a direct body orientation (in that order of importance). Other behaviours are also associated to immediacy, such as nodding, maintaining the arms open, smiling and other positive facial expressions such as happiness, interest or joy. Immediacy denotes evaluation and is related to intimacy, i.e. there is a direct and positive relationship between immediacy and Liking as well as Familiarity levels. The behavioural immediacy of a person is high with another person he/she likes and is intimate with; similarly, liking is inferred from the other's "immediate" behaviour. On average, females show a greater preference for immediacy than males.
2. **Relaxation:** Relaxation is mainly cued by the position of the trunk and of the limbs: arm-position asymmetry, openness of arm position, leg-position asymmetry, sideway lean and backward recline of the body are typical of a relaxed state. Another strong clue is the natural or tense condition of the hands and of the neck. The relaxation dimension is related to the social status of the participants: relaxation is high when a lower status individual is addressed, and it is low when the other is hierarchically superior. Dominant personality ratings correlate with relaxation, anxious and

particularly respectful individuals exhibit less relaxation. Relaxation is also related to liking: people tend to be moderately relaxed with those they like but assume very relaxed postures with those they do not respect. However, when conflicts finally arise, very tense postures are adopted. In social situations, it has been demonstrated that males are generally more relaxed than females, who tend to convey more submissive attitudes.

3. **Responsiveness**: Responsiveness, in contrast to immediacy and relaxation, is not cued by any postural characteristics, but is related to the activity level of the participants. Sweeping gestures, facial and verbal expressions in rapid succession, and every action involving a significant energy expenditure, are typical of high responsiveness. This arousal factor is correlated with extrovert and empathic personality traits, while introvert and depressive individuals have a tendency to withdraw from the interaction and exhibit low responsiveness. Men's responsiveness is often higher than women's.

Mehrabian's framework can be summarized with three metaphors (Rivano-Fischer 1988): (1) the approach metaphor: we approach what we like and we avoid what we dislike, (2) the power metaphor: we show to the people who dominate us that they are powerful and we show to the people that we dominate that they do not impress us, (3) the arousal activity metaphor: we signal our presence with increased activity levels.

6.2.1 Integration of the Theoretical Framework: Calculation of Immediacy, Relaxation and Responsiveness Levels

Our approach for integrating Mehrabian's model in our work consists in computing, using a number of simple rules, an interaction score in the three dimensions based on the individual social-psychological characteristics, the nature of the interpersonal relationship and the interaction type. We base our calculation on the indications provided by Mehrabian; for example we model the impact of Liking and Familiarity on the three dimensions. However, we consider that gender influence is already taken into account with the use of social-psychological variables such as Extraversion. We also use the assumption that the attitude of the interaction initiator is more directly linked to the interaction type than the interaction receiver's, who initially ignores the purpose of the interaction.

	Immediacy	Relaxation	Responsiveness
1. Shows solidarity	0.9	0.6	0.7
2. Shows tension release	0.75	0.8	0.8
3. Agrees	0.6	0.7	0.6
4. Gives suggestion	0.5	0.6	0.5
5. Gives opinion	0.5	0.6	0.5
6. Gives orientation	0.5	0.6	0.5
7. Asks for orientation	0.5	0.4	0.3
8. Asks for opinion	0.5	0.4	0.3
9. Asks for suggestion	0.5	0.4	0.3
10. Disagrees	0.4	0.3	0.6
11. Shows tension	0.25	0.2	0.8
12. Shows antagonism	0.1	0.4	0.7

Table 9 Default scores in Immediacy, Relaxation and Responsiveness according to the interaction type

In order to establish the influence of the interaction type on the participant's attitude, we have specified a default score for every interaction type and every dimension. These scores are given in Table 9. On the basis of this reference level and of SIML information, a score in Immediacy, Relaxation and Responsiveness can be calculated for the interaction initiator and receiver using the following formulas, given:

- the interaction initiator index $i \in \mathbf{Z}_{+}$,
- the interaction receiver index $r \in \mathbf{Z}_{+}$,
- the resulting Immediacy score $I \in \mathbf{R}$: $0.0 <= I <= 1.0$,
- the Immediacy default score for the interaction type $B \in \mathbf{R}$: $0.0 <= B <= 1.0$,
- a global parameter representing the maximum score variation related to the interaction success $V \in \mathbf{R}$: $0.0 <= V <= 1.0$,
- the interaction success $S \in \mathbf{R}$: $0.0 <= S <= 1.0$,
- the Familiarity value $F \in \mathbf{R}$: $0.0 <= F <= 1.0$,
- the Liking value $L \in \mathbf{R}$: $0.0 <= L <= 1.0$

For IPA interactions 1- 9:

$$I_{ir} = \frac{2(B + (V * S) - \frac{V}{2}) + F_{ir} + L_{ir}}{4} \qquad I_{ri} = \frac{B + (V * S) - \frac{V}{2} + F_{ri} + L_{ri}}{3}$$

For IPA interactions 10- 12:

$$I_{ir} = \frac{2(B - (V * S) + \frac{V}{2}) + F_{ir} + L_{ir}}{4} \qquad I_{ri} = \frac{B - (V * S) + \frac{V}{2} + F_{ri} + L_{ri}}{3}$$

Given:

- the resulting Relaxation score $R \in \mathbf{R}$: $0.0 <= R <= 1.0$,
- the Relaxation default score for the interaction type $B \in \mathbf{R}$: $0.0 <= B <= 1.0$,
- the Leadership value $A \in \mathbf{R}$: $0.0 <= A <= 1.0$,
- the Power value $P \in \mathbf{R}$: $0.0 <= P <= 1.0$,
- the Prestige value $E \in \mathbf{R}$: $0.0 <= E <= 1.0$,
- the status difference $D \in \mathbf{R}$: $-1.0 <= D <= 1.0$

For IPA interactions 1-3 and 10-12:

$$D_{ir} = \frac{A_i - A_r + P_i - P_r + E_i - E_r}{3} \qquad D_{ri} = \frac{A_r - A_i + P_r - P_i + E_r - E_i}{3}$$

For IPA interactions 4 and 9: $D_{ir} = P_i - P_r$ $\quad D_{ri} = P_r - P_i$

For IPA interactions 5 and 8: $D_{ir} = E_i - E_r$ $D_{ri} = E_r - E_i$

For IPA interactions 6 and 7: $D_{ir} = A_i - A_r$ $D_{ri} = A_r - A_i$

For IPA interactions 1-3 and 10-12:

$$R_{ir} = \frac{2(B + (V * S) - \frac{V}{2}) + \frac{D_{ir}}{2} + 0.5 + (1 - L_{ir})}{4} \qquad R_{ri} = \frac{B + (V * S) - \frac{V}{2} + \frac{D_{ri}}{2} + 0.5 + (1 - L_{ri})}{3}$$

For IPA interactions 4-9:

$$R_{ir} = \frac{2(B + (V * S) - \frac{V}{2}) + \frac{D_{ir}}{2} + 0.5}{3} \qquad R_{ri} = \frac{1 - (B + (V * S) - \frac{V}{2}) + \frac{D_{ri}}{2} + 0.5}{2}$$

Given:

- the resulting Responsiveness score $O \in \mathbf{R}$: $0.0 <= O <= 1.0$,
- the Responsiveness default score for the interaction type $B \in \mathbf{R}$: $0.0 <= B <= 1.0$,
- the Extraversion value $X \in \mathbf{R}$: $0.0 <= X <= 1.0$

$$O_{ir} = \frac{2(B + (V * S) - \frac{V}{2}) + F_{ir} + X_i}{4} + 0.2 \qquad O_{ri} = \frac{B + (V * S) - \frac{V}{2} + F_{ri} + X_r}{3}$$

If needed, these formulas could be easily enriched, e.g. the Immediacy score could take into account the SIML Agreeableness rating, and the Relaxation calculation could use the Stability rating.

Except when specific social norms must be taken into account, e.g. the collision avoidance mechanism directly accesses individual characteristics, we exclusively use the three scores in order to generate the participants' nonverbal behaviours, as in the case of the parametric animations that are described in Section 6.4.

6.3 Nonverbal Interactions Structure

The different aspects of nonverbal interaction that need to be simulated are precisely organized in the temporal dimension. In this section, we describe the high-level structure of nonverbal interactions in our system, i.e. we specify the order of execution of every required action. Each of these actions is described in the next sections or corresponds to one of the services already available in the integration platform, e.g. walking. We are not concerned with the nonverbal content itself, which is, as demonstrated by Kendon (1981), largely independent from the structural features of face-to-face interactions.

Three main categories of nonverbal interaction are modelled in our work: (1) Dyadic interactions, which involve two participants (2) Group interactions, which involve at least three participants and, frequently, all group members (3) Self interactions, which describes the behaviour of an individual who is temporarily not involved in any interaction with other group members.

6.3.1 Dyadic Interaction

Table 10 shows the common structure of dyadic interactions. "I" represents the interaction initiator and "R" is the interaction receiver. Automatically generated postures, adaptors and observation behaviours are described in the next section. The specification of the approach trajectory, position and orientation for interacting, and of the collision avoidance behaviour, are given in Section 6.5. The mechanisms involved in the "Interaction" phase are described in Section 6.6.

In order to determine if greeting actions are required, each agent keeps track of who it has already been interacting with. When no new dyadic interaction is waiting to be performed, a Self interaction is automatically created for every participant.

Phase	Event	Action	Preparation
Initialisation	I: a message stating that a new dyadic interaction must be performed is received	I: initiation of observation behaviour	I: calculation of a suitable trajectory to approach R
Initial approach		I: walk toward R while avoiding collisions	
Perceptual contact	R: when I is close enough, R perceives his/her presence	R: initiation of observation behaviour	I: calculation of the position and orientation for interacting R: calculation of the orientation for interacting
Final approach		I: walk to the position for interacting R: modification of its orientation	
Interaction initialisation		I: if required, the orientation is adjusted; if required, a greeting action is triggered; a posture is generated R: if required, a greeting action is triggered; a posture is generated	
Interaction		I: various gestures and facial expressions are performed, including adaptors R: acknowledging gestures and facial expressions are performed, as well as adaptors	
Interaction termination	I and R: an interaction termination signal is received I and R: a new interaction signal may be received	I and R: check if a new interaction involving the same participants is received; if not, a greeting action is triggered, then I and R face different locations R: if R is the initiator of a new interaction, a turn-taking action is triggered	I and R: back to the Interaction phase or preparation of a Self interaction

Table 10 Dyadic interactions structure

6.3.2 Group Interaction

Contrary to what occurs in dyadic interactions, group interaction receivers move toward the group interaction initiator, as can be seen in Table 11. The approach phase is somewhat simplified since receivers are directly oriented to the initiator, and the initiator is oriented to the group as a whole. However, the initiator's observation behaviour must be here regularly directed at another participant. As it is the case with dyadic interactions, every participant enters a period of Self interaction when the flow of standard interactions is temporarily suspended.

Phase	Event	Action	Preparation
Initialisation	I: a message stating that a new group interaction must be performed is received	I: walk to a central location, an attention-getting action is performed	I: calculation of an average orientation for addressing the group
Perceptual contact	R: perception of the attention-getting action	I: modification of its orientation	R: calculation of the position for interacting
Approach		R: walk toward I while avoiding collisions	
Interaction initialisation		I: if required, a greeting action is triggered; initiation of observation behaviour and generation of a posture R: if required, a greeting action is triggered; initiation of observation behaviour and generation of a posture	
Interaction		I: various gestures and facial expressions are performed, including adaptors R: acknowledging gestures and facial expressions are performed, also adaptors	
Interaction termination	I and R: a group interaction termination signal is detected I and R: a new group interaction signal may be received	I and R: check if a new group interaction involving the same participants is received; if not, a greeting action is triggered, then I and R face different locations R: if R is the initiator of a new interaction, a turn-taking action is triggered	I and R: back to the Interaction phase or preparation of a Self interaction

Table 11 Group interactions structure

6.3.3 Self Interaction

Few outward signs are linked to Self interactions. We have included three types of behaviours which are adopted as long as the group member is not involved in any dyadic or group interaction:

- A number of adaptor gestures are regularly performed (using the SIML Stability rating as Relaxation parameter, as described in the next section).

- Facing actions are executed (using random coordinates).
- Observation behaviours are adopted (using random coordinates).

6.4 Specialized Motion Synthesis

In order to facilitate future integration, the work we present in this chapter is implemented in the VHD++ development framework (Ponder et al. 2003). The technology included in VHD++ is identical to the one presented in Chapter 3 for face animation, and, as it will appear, it is also very similar for body animation. VHD++ provides a number of low and high-level services for the real-time animation of H-ANIM compliant virtual characters, among others:

- the ability to modify the position and orientation of the characters in a scene,
- the ability to manipulate individual joint angles through a library called HBODY,
- the ability to play predefined body animations in the WRK format,
- the ability to animate faces using predefined facial expressions specified with FAPs, as well as the ability to blend several expressions together,
- a walk engine allowing the use of different walking styles

In addition, we have developed a general-purpose Facing action which is similar to the one developed for AML. It gradually modifies the global orientation of a character according to provided coordinates, while playing in loop a keyframe sequence representing a right or left side step.

Parametric animations, such as Facing or Walking, greatly facilitate the simulation of realistic nonverbal interactions, since they constitute a flexible way of controlling the character's behaviour. Motion synthesis is made on the basis of a few high-level parameters, and the update of the character can take into account its current posture, position or orientation. In contrast, predefined animations cannot be as finely adapted to the context and may alter some pre-existing aspects of the virtual character which need to be preserved. Therefore, in addition to the functionalities already included in VHD++, it is crucial to provide a number of specialized parametric animations replicating some important and specific aspects of nonverbal communication.

6.4.1 Parametric Postures

A posture corresponds to a position of the body and of its parts, compared to a determined system of references. Postures are particularly efficient at conveying interest, attention, the lack of interest or attention, positive or negative evaluation and feelings. Mehrabian (1972) proposes a two dimensional framework for the characterization of postural cues based on Immediacy and Relaxation. Logically, Responsiveness is not involved here since it is related to the interaction's temporal dimension. As previously mentioned, a forward rather than a backward lean of the body is typical of high Immediacy, and an asymmetrical rather than a symmetrical position of the limbs signals Relaxation; Mehrabian even provides weights characterizing the influence of each part of the body compared to the others.

On the basis of these findings, we have implemented a motion synthesis function which updates the rotation matrix controlling the orientation of a number of joints in a character's virtual skeleton, according to a score in Immediacy and Relaxation. Table 12 contains the list of involved joints[8] and rotation axis, as well as the rotation angles in radian which are applied to the joints in order to produce an adequate posture. For each joint, a delta angle proportional to the Immediacy and Relaxation score is calculated on the basis of the provided ranges: the minimum value is used when the Immediacy or Relaxation score is equal to 0, the maximum value when it is equal to 1, and, when the score is 0.5, the delta angle is always null.

Body asymmetry can be produced in several ways, e.g. moving the head to the left or to the right, raising the left arm or the right arm, etc. Therefore, we have included a mechanism which alternatively selects different rotation directions and symmetric joints in order to produce the desired effect. In Table 12, the numbers in parenthesis are alternative range values used by this mechanism. Various postures expressing the same idea are thus generated.

Joint	**Axis**	**Relaxation: delta angle range**		**Axis**	**Immediacy: delta angle range**	
		Min	**Max**		**Min**	**Max**
SKULLBASE	X Y	Π/4.0 0.0	-Π/8.0 Π/8.0 (-Π/8.0)	X	-Π/16.0	Π/16.0
VL5	Z	0.0	Π/16.0 (-Π/16.0)	X	-Π/16.0	Π/5.0
R_HIP	Y	Π/8.0	-Π/32.0 (-Π/3.0)			
L_HIP	Y	-Π/8.0	Π/3.0 (Π/32.0)			
R_KNEE	X	0.0	0.0 (Π/6.0)			
L_KNEE	X	0.0	Π/6.0 (0.0)			
R_ANKLE	Y	0.0	0.0 (-Π/2.0)			
L_ANKLE	Y	0.0	Π/2.0 (0.0)			
R_SHOULDER	Y Z	Π/2.0 0.0	-Π/2.0 (-Π/16.0) 0.0 (-Π/10.0)			
L_SHOULDER	Y Z	-Π/2.0 0.0	Π/16.0 (Π/2.0) Π/10.0 (0.0)			
R_ELBOW	X Y	0.0 0.0	0.0 (-Π/4.0) 0.0 (Π/6.0)			
L_ELBOW	X Y	0.0 0.0	-Π/4.0 (0.0) -Π/6.0 (0.0)			

Table 12 Joints and angles used in the generation of parametric postures

The following algorithm, which is applied to every involved joint, is used in order to generate a new posture:

1. The rotation angles to apply on every axis are calculated on the basis of Immediacy and Relaxation scores, using linear interpolation in the range 0→Min or 0→Max.
2. A full rotation matrix is built with these angles.
3. The initial joint matrix is multiplied by the rotation matrix in order to obtain the new final orientation of the joint.

[8] The numerous joints of the hands, which gradually open as relaxation grows, have not been included in the table.

4. Quaternion interpolation is used for the calculation of the intermediate matrices between the initial orientation and the new final orientation, in order to generate a smooth transition.

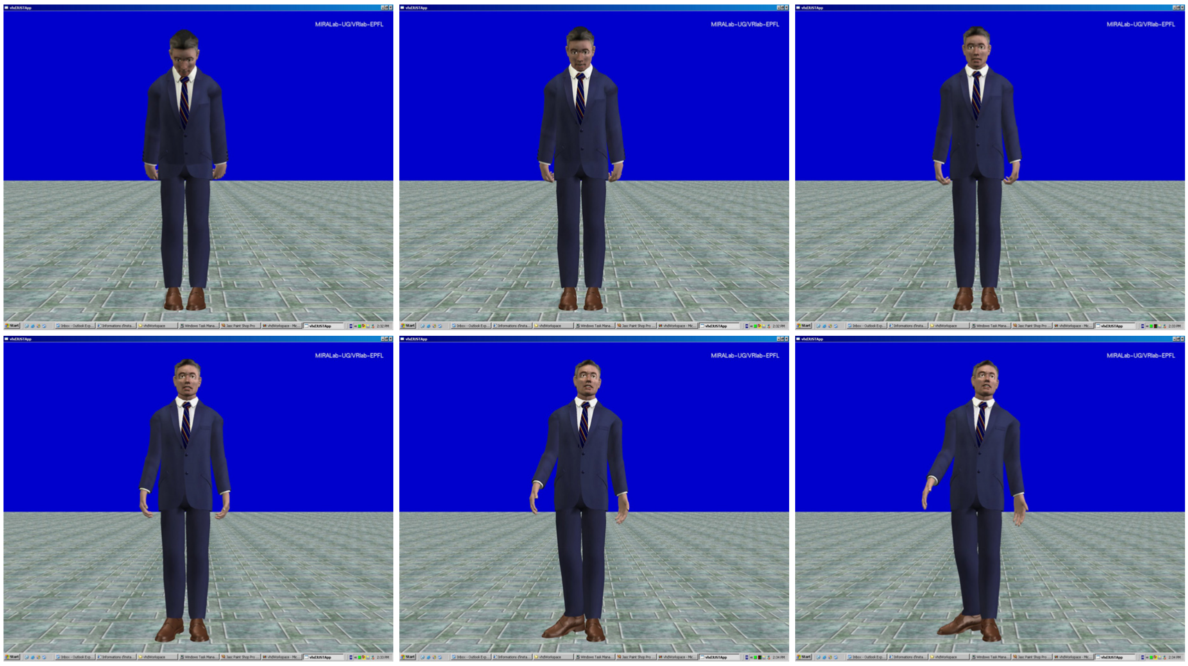


Figure 61 Examples of parametric postures: from low Relaxation and high Immediacy to high Relaxation and low Immediacy

To produce a consistent visual outcome, we generate, in addition to the posture itself, a facial expression matching the parameters given as input. For this purpose, we take advantage of the high-level FAPs described in Chapter 3, which correspond to the six basic facial expressions identified by Ekman and Friesen (1969). We also use a number of functions provided in VHD++ allowing the blending of several expressions and the control of their intensity.

When a new posture is built, two facial expressions are automatically created and blended: one for the Immediacy dimension, another for the Relaxation dimension. Table 13 shows the selected parameters, based on the "Anger", "Fear" and "Joy" expressions.

	Relaxation < 0.5	**Relaxation >= 0.5**
Immediacy < 0.5	*Expression1="Anger"* *Intensity=2(0.5-Immediacy)* *Expression2="Fear"* *Intensity=2(0.5-Relaxation)*	*Expression1="Anger"* *Intensity=2(0.5-Immediacy)* *Expression2="Joy"* *Intensity=2(Relaxation-0.5)*
Immediacy >= 0.5	*Expression1="Joy"* *Intensity=2(Immediacy-0.5)* *Expression2="Fear"* *Intensity=2(0.5-Relaxation)*	*Expression1="Joy"* *Intensity=2(Immediacy-0.5)* *Expression2="Joy"* *Intensity=2(Relaxation-0.5)*

Table 13 Parameters for the generation of facial expressions based on Immediacy and Relaxation

For example, a participant with a high score in Immediacy but a very low score in Relaxation, will display a facial expression which is a mix of joy and fear. Since Ekman and Friesen identified joy as the only basic expression conveying positive feelings, it is used for high levels in both Immediacy and Relaxation. Therefore, in order to display a "pure" expression of joy, a participant must have a maximum score in Immediacy and in Relaxation.

Figure 62 A facial expression of low Immediacy and high Relaxation

Figure 63 A facial expression of high Immediacy and low Relaxation

Compared to the use of predefined postures, our approach has a number of important advantages:

- The state of mind of the characters can be matched very closely.
- The design overhead is reduced: since two dimensions are simultaneously used, it would be necessary to prepare a very high number of predefined sequences in order to provide a similar level of characterization.
- Although the best results are produced when the function is applied to the default posture, the pre-existing skeleton configuration is partially preserved since a limited number of joints are updated: those which are particularly efficient at conveying meaning.
- A variety of postures conveying the same meaning can be generated without additional work.

Although a formal evaluation would be required in order to determine the exact level of realism of the produced visuals, it was fascinating to observe how the introduction of Mehrabian's parameters in the function gradually induced the formation of meaningful postures.

6.4.2 Parametric Adaptors

As underlined in Chapter 2, people never keep totally still in the context of social interaction and even very small movements convey some meaning. Ekman and Friesen (1969) call "adaptors" these actions like small movements of the wrists and of the legs, scratching or playing with some body parts, facial tics and other specific expressions, that we continually perform in an unconscious way. Generally, adaptors are considered a display of discomfort. In the context of small groups, it has been established (Burgoon, Buller & Woodall 1996) that adaptors are used in the inverse proportion to the status of the members, e.g. few adaptors are observed in the nonverbal behaviour of group leaders. We rely on the Relaxation score to characterize adaptors, since this dimension comprises the notion of comfort and is based on status differences.

In order to simulate this aspect of nonverbal behaviour, we use a combination of direct joint manipulation and predefined animations. The parametric animation function we have implemented continually introduces small variations in the joint angle values of the character's virtual skeleton, and, occasionally, predefined animations representing more complex gestures (e.g. scratching its head) or facial expressions (e.g. winking and blinking) are triggered. When it is important to preserve the character's posture, only the first part of the mechanism is active. In this case, the joints' angle values oscillate around their initial orientation, thus the character's overall appearance and its associated meaning are not corrupted.

Two dimensions are affected by the Relaxation score: the frequency and the speed of the adaptors. An adequate adaptor frequency is generated by calculating, after each performed action, a countdown value for the triggering of the next action. Given:

- the resulting countdown value in number of frames $C \in \mathbf{Z}_+$,
- the Relaxation score $R \in \mathbf{R}$: $0.0 <= R <= 1.0$,
- a global parameter representing the minimum pause duration in seconds between two actions $P \in \mathbf{R}_+$,
- the number of frames per second $F \in \mathbf{Z}_+$,
- a random value $D \in \mathbf{R}$: $0.0 <= D <= 1.0$

For single joint manipulations: $C = PF + 5(PF\frac{R+D}{2})$

For predefined body animations: $C = 10PF + 50(PF\frac{R+D}{2})$

For predefined facial animations: $C = 3PF + 15(PF\frac{R+D}{2})$

The speed of the animation is calculated as follows, given:

- the resulting animation speed $S \in \mathbf{R}_+$

$$S = 0.5 + 2\frac{1-R}{2}$$

The method for updating individual joints is identical to the one described for parametric postures, except that a random score is used to determine the delta angle that must be applied to the joints. The average random score must be equal to 0.5 in order for the posture to be preserved. Table 14 shows the joints updated in our implementation. Since no detailed information is available about the joints and angles involved in the typical adaptors (small wrists rotation and legs movements are generally mentioned), the provided data is the result of our initial tests.

Joint	Axis	Delta angle range	
		Min	Max
R_WRIST	Y	-Π/3.0	Π/3.0
L_WRIST	Y	Π/3.0	-Π/3.0
VL5	Z	-Π/32.0	Π/32.0
R_KNEE	X	0.0	Π/12.0
L_KNEE	X	0.0	Π/12.0
R_ANKLE	Y	Π/2.0	-Π/2.0
L_ANKLE	Y	-Π/2.0	Π/2.0
R_ELBOW	X Y	0.0 -Π/6.0	-Π/8.0 Π/6.0
L_ELBOW	X Y	0.0 Π/6.0	-Π/8.0 -Π/6.0

Table 14 Joints and angles used in the generation of parametric adaptors

6.4.3 Parametric Observation

According to Mehrabian (1972), observation behaviours in general and eye contact in particular are fundamental aspects of nonverbal communication. In real life interactions, participants do not constantly look toward the other but alternate between periods of direct observation and periods of indirect observation in which they look away. Mehrabian has shown that the total duration of direct observation in an interaction is proportional to the degree of affiliation of the participants, and, therefore, to Immediacy. It can be said that people look at what they like and avoid looking at what they dislike. Eye contact is directly related to the observation behaviour, since it occurs when both interactants look at each other at the same instant. Thus, the frequency of eye contacts is affected by the direct observation ratio of both participants. These factors must be taken into account in the development of an observation function that allows virtual characters to stare at each other while their bodies are not directly aligned.

Our parametric observation function controls the SKULLBASE joint of a virtual character[9]. It takes as input three coordinates that represent the position of the interlocutor's face in 3D space, the Immediacy score and the Relaxation score. Here are the successive operations which are executed:

1. On the basis of the received coordinates and of the current body orientation, a rotation matrix *M*, corresponding to the orientation of the head toward the interlocutor's face, is composed.
2. Using quaternions, the pre-existing SKULLBASE matrix is progressively interpolated until it is equal to *M*.
3. After a given number of frames *C,* a new orientation matrix *M'* is calculated by adding a random amount of rotation around the X and Y axis to *M*; if the new rotation exceeds the SKULLBASE joint limits, another solution is computed.
4. If indirect observation is required at this stage, the SKULLBASE matrix is interpolated until it is equal to *M'*; if direct observation is required at this stage, the

[9] Ideally, a combination of skullbase and eyeballs movements would have been used; unfortunately, at the time of development, it was not possible to directly manipulate the eyeballs orientation in VHD++.

SKULLBASE matrix is interpolated until it is equal to *M* (which has no effect if it is already equal to *M*).

5. Back to step 3.

The process continues in this manner until new parameters are provided. What remains to be specified is the calculation of the number of frames between two head movements *C*, and the criterion used to decide if the new head movement needs to be oriented to the interlocutor's face or to an alternative location. We consider that regular head movements are of the same nature as adaptors, and therefore, we use the Relaxation score in order to compute *C*. Given the parameters described in the previous section:

$$C = 2PF + 10(PF \frac{R + D}{2})$$

The speed of head movements (set by changing the interpolation step) is also dependant on Relaxation, and is calculated in the same way as the speed of adaptors.

In order to decide of the orientation of the head, a variable is incremented every frame when the SKULLBASE matrix is equal to *M*. With this way of calculating the time spent directly watching the interlocutor, the transition phases have the status of indirect observation. The ratio of direct observation frames to the total number of frames can then be compared to the provided Immediacy score. Given:

- the number of direct observation frames $D \in \mathbf{Z}_{+}$,
- the total number of elapsed frames $T \in \mathbf{Z}_{+}$,
- the Immediacy score $I \in \mathbf{R}$: $0.0 <= I <= 1.0$

If $\frac{D}{T} \geq I$ is true, then M' is selected as the new orientation target; if it is false, *M* is chosen. Since this test is done only when a new head movement is about to be executed, the proportion of direct observation frames can temporarily surpass the desired level. The random value integrated in the calculation of the duration between two head movements introduces some variation also in the sequences of selected orientations.

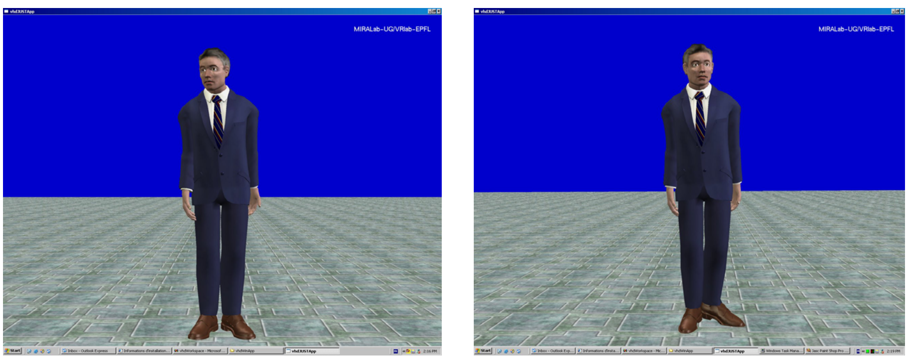


Figure 64 Example of observation behaviour with adaptors (see left leg and left wrist)

6.5 Simulation of Proxemics

Proxemics is one of the most important aspects of nonverbal communication. In this section, we explain how it is simulated in our work on the basis of psychological and anthropological findings.

6.5.1 Interpersonal Distance

How far one individual is from the other when interacting follows well established patterns. Hall's (1966) conceptualisation of "personal space" accounts for the observed regularities in this dimension of nonverbal behaviour. Personal space is an invisible area surrounding a person's body which limits the amount of physical contact and into which the presence of others generates discomfort. When their personal space is violated, people generally move away and express irritation through defensive gestures and postures. The personality and relationship of the individuals participating in the interaction as well as environmental factors (e.g. the available space) influence the size of personal spaces. As an anthropologist, Hall also underlines the fact that different cultures use different personal spaces. For western societies, he identifies four types of interpersonal distances selected by people to carry out their interactions:

1. **Intimate distance**: Intimate distance ranges from body contact to approximately 46 centimetres[10]. It is selected by persons who are very familiar and appreciate each other, like couples or close members of a family. The use of this interactional distance is generally restricted to private environments. Hall distinguishes between a close zone (0-15 cm), where intimate activities requiring physical contact can take place (e.g. arm wresting), and a far zone (15-46 cm) which is chosen when the interaction is not as touch-oriented.
2. **Personal distance**: Personal distance ranges from 46 cm to 122 cm. This type of interactional distance is the most frequently observed in small "social" groups. It is typically used in interactions such as discussions between friends and other fellow group members in an informal context. Interactions between intimates are also characterized by the use of such distances when they occur in public places. Isolated physical contacts can take place in the close zone (46-76 cm), but the far zone (76-122 cm) does not allow them, since the distance is bigger than an arm's length.
3. **Social distance**: Social distance ranges from 122 cm to 366 cm. It is used for impersonal business and routine interactions in task-oriented groups as well as in the interactions of acquaintances and strangers. It is also selected when one individual addresses a small group of people. Its close zone (122-213 cm) is adequate for informal situations while the far zone (213-366 cm) is required for more status-related interactions.
4. **Public distance**: Public distance begins at 366 cm. It is appropriate for formal meetings such as academic presentations and for highly impersonal situations like theatre, sports events, etc.

The Intimate and Public distances are clearly not relevant to interactions taking place in the context of small groups, in contrast to the Personal and Social distances which are well suited to the positioning of our characters. For dyadic interactions, we calculate interpersonal

[10] We have converted to metric measurements the distances provided by Hall.

distances ranging from Personal distance/close zone to Social distance/far zone. For group interactions, we only use Social distances. Figure 65 shows the selected ranges in proportion to our characters' representation.

In our theoretical framework, Immediacy is the dimension which accounts for interpersonal distance, since it is directly related to the perceptual availability of one participant to another. In addition, we make the assumption that perceptual availability is a function of the occupied proportion of the participants' field of view, and that it decreases within short distances at a higher rate than at greater distances. Therefore, our calculation of interaction distances uses a quadratic function based on the Immediacy score. Given:

- the resulting distance in centimetres $D \in \mathbf{R}_+$,
- the Immediacy score $I \in \mathbf{R}$: $0.0 <= I <= 1.0$

For dyadic interactions: $D = 46 + 320(1 - I)^2$ *For group interactions:* $D = 122 + 244(1 - I)^2$

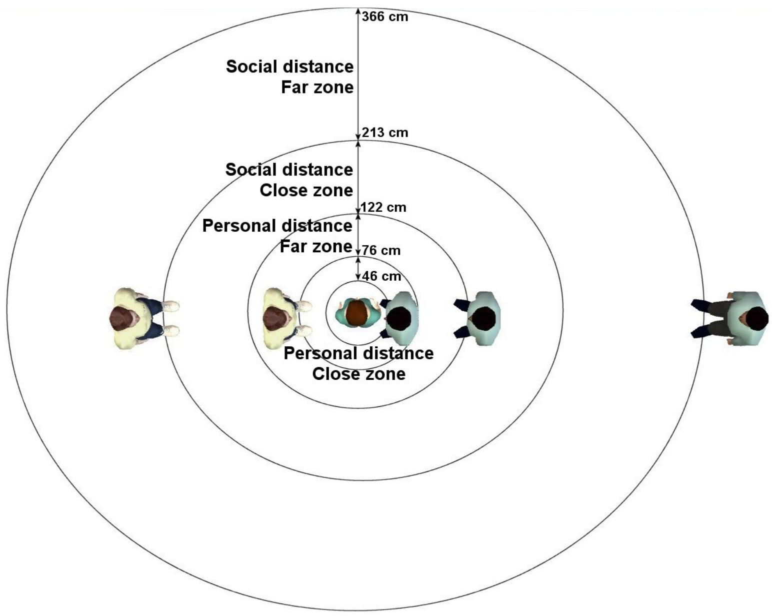


Figure 65 Interpersonal distances

6.5.2 Body Orientation

Body orientation is the other main aspect of proxemic choices. Hall, in addition to his study of interactional distances, proposes a classification of the physical arrangements between two interacting individuals. However, we base our work on the more recent contribution by Ciolek and Kendon (1980) which expands Hall's categories.

Ciolek and Kendon distinguish three main types of formation: (1) M-formations which are moving formations; two or more individuals navigate together in a given environment while maintaining interpersonal integrity. The work by McPhail, Powers and Tucker (1992), referenced in the literature review, deals with this kind of formation. (2) E-formations which are element formations. Typically, these side-by-side formations are used when the participants' attention is not mainly directed to each other but to the control or observation of some external spaces. (3) F-formations which are face-to-face formations. In these formations, the participants arrange themselves in order to have a preferential access to each other for interacting. We are concerned with the latter type of formation.

F-formations are divided into several zones: (1) a nucleus composed of an open space (o-space), a participant space (p-space) and a regional space (r-space) (2) a larger unspecified area composed of different spaces allowing different degrees of sensory processing (a-space, b-space, c-space). Ciolek and Kendon (1980) provide many structural details; the most important aspect for our work is the description of the arrangements themselves and of their associated body orientations. 26 possible arrangements have been identified. Of these 26 arrangements, 6 cover most interactions that take place on a daily basis:

- **N arrangement**: participants face each other but are positioned on the side,
- **H arrangement**: participants face each other directly,
- **V arrangement**: the frontal planes of the participants' bodies intersect outside the formation at an angle of 45 degrees,
- **L arrangement**: the frontal planes of the participants’ bodies intersect at right angle,
- **C arrangement**: participants stand at an open angle of about 135 degrees,
- **I arrangement**: participants are oriented in the same direction

Figure 66 shows the arrangements as they are simulated in our system.

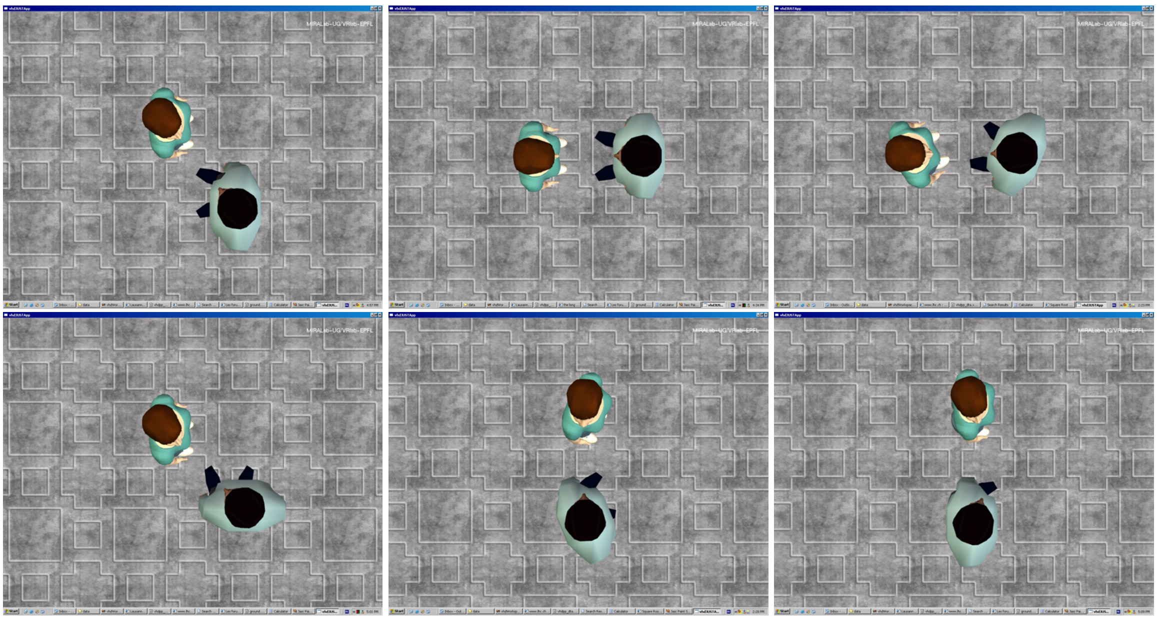


Figure 66 Simulated F-formations: (from the upper left to the lower right) N arrangement, H arrangement, V arrangement, L arrangement, C arrangement and I arrangement

Of these 6 arrangements, 3 are closed arrangements (N, H, V) and 3 are open arrangements (L, C, I). Closed arrangements isolate the dyad from the external world which lowers the probability that the interaction be disturbed, while open arrangements allow participants to direct some of their attention toward the surrounding environment. The choice of an arrangement is related to the environmental constraints but also to the participant's relationship and interaction type. Our approach for the selection of arrangements is to favour open arrangements for low Immediacy levels, and closed arrangements for high Immediacy levels. Since we do not model environmental constraints, a simple random value is used to represent this dimension. In addition, we take into account the observation (Corraze 1980) that open conflicts and other high tension situations take place while both participants are directly oriented to each other as a sign of hostility. Here is the simple rule used, given:

- the participant Immediacy score $I \in \mathbf{R}$: $0.0 <= I <= 1.0$,
- the interaction Immediacy score $I' \in \mathbf{R}$: $0.0 <= I' <= 1.0$,
- the Relaxation score $R \in \mathbf{R}$: $0.0 <= R <= 1.0$,
- a random value $A \in \mathbf{R}$: $0.0 <= A <= 1.0$

$$I' = \frac{I_i + I_r + A}{3}$$

If $I' > \frac{5}{6}$ *or* $\frac{R_i + R_r}{2} < 0.3$: *H arrangement is selected*

else if $I' > \frac{4}{6}$: *N arrangement is selected, else if* $I' > \frac{3}{6}$: *V arrangement is selected,*

else if $I' > \frac{2}{6}$: *L arrangement is selected, else if* $I' > \frac{1}{6}$: *C arrangement is selected*

else: I arrangement is selected

To the best of our knowledge, no similar data is available for group interactions. In these conditions and given the fact that personal involvement is by nature lower in group interactions, we hypothesize that the "risk" taken by the participants adopting a direct orientation toward the initiator is acceptable in group interactions. In our simulations, receivers are therefore simply oriented toward the initiator and an average orientation is calculated on the basis of the position of every participant in order for the initiator to face the group.

6.5.3 Interaction Position and Approach

6.5.3.1 Dyadic Interactions

Using the selected interaction distance and arrangement, adequate interaction position and approach trajectory need to be computed. In order for the approach to appear as natural as possible, at least two patterns should be avoided:

- In most cases, the initiator must not walk straight to the receiver, since this approach trajectory, like it is the case for direct orientation in dyadic standing encounters, can be interpreted as a sign of high familiarity or aggression.
- The amount of rotation of the facing action which is used when the character is stationary must be limited since people are known to anticipate their future positioning and to smooth direction changes (Musse & Thalmann 1998).

In order to take into account these aspects, we identify two important locations in the area separating the participants: (1) the initiator's final position for interacting *f*, (2) an approach point *a* that the initiator must reach before arriving to the receiver. The final position is situated on the straight line between the two participants, at a distance of the receiver corresponding to the calculation given in Section 6.5.1. Given:

- the absolute angle between the two participants $\rho \in \mathbf{R}$: $0.0 <= \rho <= 2\Pi$,
- the initial distance between the two participants in centimetres $G \in \mathbf{R}_{+}$,
- the interaction distance between the two participants in centimetres $D \in \mathbf{R}_{+}$,
- a coordinate $X \in \mathbf{Z}$,
- a coordinate $Z \in \mathbf{Z}$

$$\rho = \tan^{-1}\left(\frac{Z_i - Z_r}{X_i - X_r}\right) \qquad G = \sqrt{(X_i - X_r)^2 + (Z_i - Z_r)^2}$$

$$X_f = (X_i + G - D)\cos(\rho) \qquad Z_f = (Z_i + G - D)\sin(\rho)$$

We consider that both participants contribute equally to establishing suitable interaction angles. Table 15 provides the angles relative to a direct orientation of the participants, which generate the desired arrangements. The angles in parenthesis correspond to the inverse rotation; they are selected when they represent the minimum energy solution.

Arrangement	**Initiator's delta angle**	**Receiver's delta angle**
N	Π/4.0 (-Π/4.0)	Π/4.0 (-Π/4.0)
H	0.0	0.0
V	Π/8.0 (-Π/8.0)	-Π/8.0 (Π/8.0)
L	Π/4.0 (-Π/4.0)	-Π/4.0 (Π/4.0)
C	3Π/8.0 (-3Π/8.0)	-3Π/8.0 (3Π/8.0)
I	Π/2.0 (-Π/2.0)	-Π/2.0 (Π/2.0)

Table 15 Rotation angles used to generate F-formations

On the basis of the coordinates of the initiator's final position and of the angle corresponding to the selected arrangement, we can calculate the approach point's location.

Given:

- the initiator's delta angle corresponding to the selected arrangement (found in Table 15) $\alpha \in \mathbf{R}$

$$X_a = (X_f - 366 + D)\cos(\rho - \alpha) \qquad Z_a = (Z_f - 366 + D)\sin(\rho - \alpha)$$

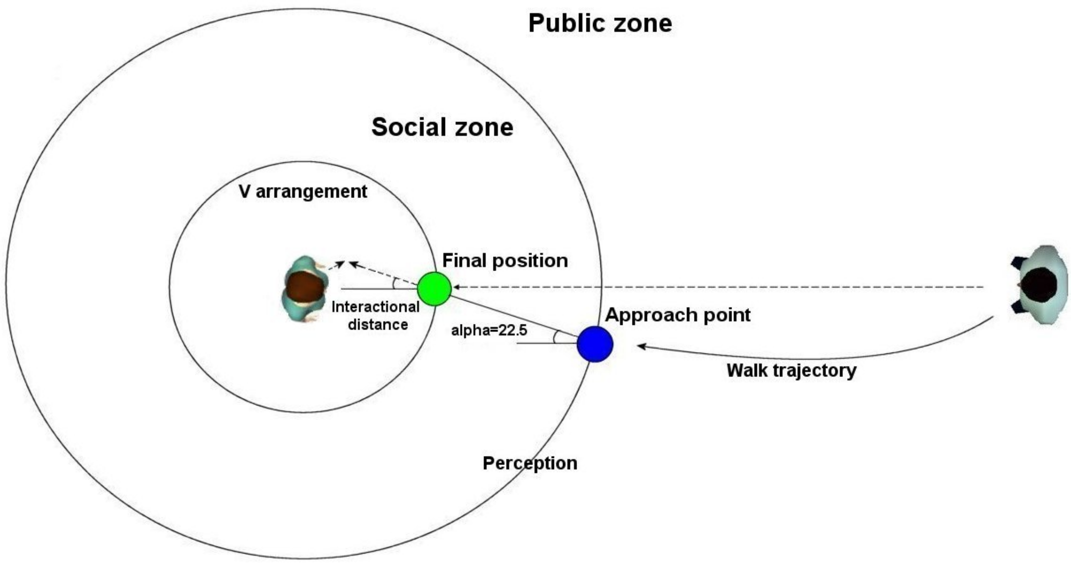


Figure 67 Approach for a V arrangement

As it can be seen in Figure 67, the approach point is situated at the beginning of the receiver's public zone (366 cm). We consider that the initiator is perceived and acknowledged when he/she reaches this distance. Final adjustments using the facing action handle the part of the rotation exceeding Π/4.0 (or -Π/4.0) in order to avoid unnatural trajectories, as well as the possible error margin, e.g. when the final position is very close to the approach point. By setting differently the maximum angular velocity used by the walk engine, the final orientation error margin can be lowered or the trajectory can be further smoothed.

The orientation of the receiver is also calculated on the basis of the delta angles provided in Table 15 and is adopted by triggering a Facing action.

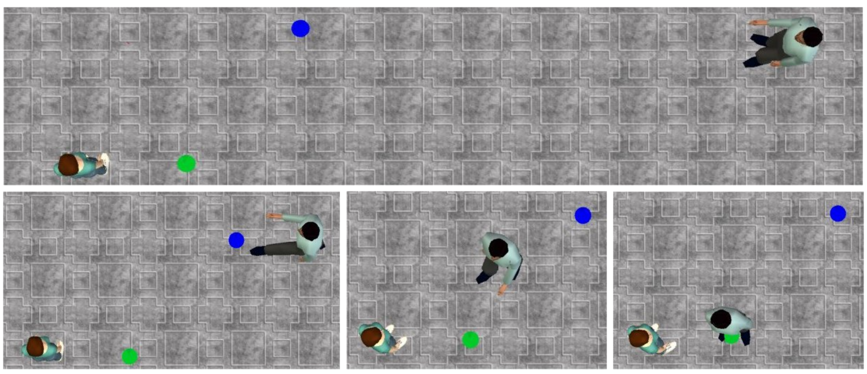


Figure 68 Example approach

6.5.3.2 Group Interactions

In contrast with dyadic interactions participants, receivers involved in our group interactions directly walk to their final position. However, the calculation of this position is here more complex since their relationship with the other receiver participants must also be taken into account. For this purpose, we use a spring embedder algorithm similar to the one described in Chapter 5, but situated in 2D space instead of 3D space. The use of this technique amounts to compute physical forces in order to represent social forces. Three initial steps are required:

1. An Immediacy score is calculated for every participant's relationship as described in Section 6.2.1 except that the interaction type must not be taken into account for receiver-receiver relationship (a neutral interaction default score of 0.5 is used).
2. The desired interpersonal distances are calculated on the basis of these Immediacy scores and of the formulas presented in Section 6.5.1; they are provided to the spring embedder routine.
3. The current position (X and Z coordinates) of every involved participant is provided to the spring embedder routine.

Once the energy of the system has been minimized, the position of every participant can be extracted. Typically, because of the fact that receiver-receiver distances are not affected by the interaction itself, the initiator will appear at the centre of the group when he/she triggers a positive social-emotional interaction, and at the periphery for negative interactions.

This approach ensures that every participant has its own non-overlapping position and that every interpersonal relationship is properly taken into account. However, depending on the constraints, a limited difference between the desired distances and the actual distances between the calculated positions, may be observed.

6.5.4 Socially Intelligent Collision Avoidance

Collision avoidance behaviours are related to Proxemics since they are linked to the issue of personal space and convey meaning. Several models exist to solve the problem from a purely geometrical point of view, but the social nature of these behaviours is rarely taken into account. In this section, we present our approach of the socially intelligent avoidance of collisions for walking characters. The first part describes our implementation of a low-level collision avoidance mechanism which is based on Feurtey's (2000) work but includes several additions. The second part explains how the low-level mechanism can be socially customized. We do not consider the problem of path planning in complex physical environments, which is not directly relevant to our topic.

6.5.4.1 Collision Avoidance Mechanism

A collision avoidance mechanism involves two main tasks: (1) the detection of potential collisions (2) the identification of a suitable avoidance strategy.

Although walking takes place in 2D space *(X,Z)*, our mechanism operates in 3D space *(X,Z,T)*: in order to detect potential collisions, the future positions of every character are predicted every timeframe on the basis of its initial position, destination and speed. The temporal dimension's unit is the frame number relative to the current frame. Depending on the requirements, the number of frames that are processed can be customized, thus extending or shortening the character's "depth of view". With such an approach, the problem is greatly simplified since every object becomes static.

In order to determine if a trajectory is valid, it is also necessary to calculate the characters' interpersonal distance with every other perceived character as described in the preceding section, i.e. without taking into account the interaction type. The presence of another character within the calculated range represents a kind of "social collision". On the basis of these distances and of the prediction of every character's position, it can be easily determined if a collision is going to occur: the predicted positions are scanned from frame number 1 to frame number *n*, the distance between the characters is calculated at each frame, this distance is compared to the minimum allowed distance. If it is lower, an alternative trajectory must be selected.

Three types of corrective actions can be used to avoid collision:

1. Detouring: it involves a clockwise or anti-clockwise orientation change.
2. Accelerating: in the problem's 3D space, it corresponds to passing under (i.e. earlier) the collision point.
3. Decelerating: in the problem's 3D space, it corresponds to passing above (i.e. later) the collision point.

In order to select a new trajectory, a number of candidate solutions involving different orientation and speed modifications are evaluated. A global rating composed of several individual costs is calculated for every solution. These costs can be physical costs linked to energy expenditure, but, as it will appear, may also be of a different nature. The first cost we consider is the cost of changing direction, given:

- the resulting cost of changing direction $C_1 \in \mathbf{R}$: $0.0 <= C_1 <= 1.0$,
- a weight parameter for the cost of changing orientation $W_1 \in \mathbf{R}$: $0.0 <= W_1 <= 1.0$,
- the current orientation angle about the Y axis $O \in \mathbf{R}$: $0.0 <= O <= 2\Pi$,
- the candidate orientation angle about the Y axis $O' \in \mathbf{R}$: $0.0 <= O' <= 2\Pi$

$$C_1 = W_1 \frac{|O - O'|}{\Pi}$$

When the candidate walking speed is higher than the current speed, a cost of acceleration is computed, given:

- the resulting cost of acceleration $C_2 \in \mathbf{R}$: $0.0 <= C_2 <= 1.0$,
- a weight parameter for the cost of acceleration $W_2 \in \mathbf{R}$: $0.0 <= W_2 <= 1.0$,
- the current speed in meters per second $S \in \mathbf{R}_+$,
- the candidate speed in meters per second S' axis $\mathbf{R}_+$,
- the minimum allowed speed in meters per second S_{min} $\mathbf{R}_+$,
- the maximum allowed speed in meters per second S_{max} $\mathbf{R}_+$

$$C_2 = W_2 \frac{S' - S}{S_{\max} - S_{\min}}$$

When the candidate walking speed is lower than the current speed, a cost of deceleration is computed, given:

- the resulting cost of deceleration $C_3 \in \mathbf{R}$: 0.0 <= C_3 <= 1.0,
- a weight parameter for the cost of deceleration $W_3 \in \mathbf{R}$: 0.0 <= W_3 <= 1.0

$$C_3 = W_3 \frac{S - S'}{S_{\max} - S_{\min}}$$

Although we want our characters to avoid collisions, they must still reach their destination. Therefore, we introduce a cost of moving away from the goal, given:

- the resulting cost of moving away from the goal $C_4 \in \mathbf{R}$: 0.0 <= C_4 <= 1.0,
- a weight parameter for the cost of moving away from the goal $W_4 \in \mathbf{R}$: 0.0 <= W_4 <= 1.0,
- the goal's coordinates $X_g, Z_g \in \mathbf{N}$,
- the coordinates reached after n frames using a directly goal orientated angle $X, Z \in \mathbf{N}$,
- the coordinates reached after n frames using the candidate orientation $X', Z' \in \mathbf{N}$

$$C_4 = 1 - W_4 \frac{\sqrt{(X - X_g)^2 + (Z - Z_g)^2}}{\sqrt{(X' - X_g)^2 + (Z' - Z_g)^2}}$$

In Feurtey's algorithm, the character's walking speed remains constant once it has been modified to avoid a collision. In our simulations, a different speed can be selected according to the interaction type, e.g. a fast walk for a "Shows tension" interaction, and it should be restored after that the collision has been avoided. Therefore, we introduce a cost of speed inadequacy, given:

- the resulting cost of speed inadequacy $C_5 \in \mathbf{R}$: 0.0 <= C_5 <= 1.0,
- a weight parameter for the cost of speed inadequacy $W_5 \in \mathbf{R}$: 0.0 <= W_5 <= 1.0,
- the interaction speed in meters per second $S \in \mathbf{R}_+$,
- the candidate speed in meters per second $S' \in \mathbf{R}_+$

$$C_5 = W_5 \frac{|S' - S|}{S_{\max} - S_{\min}}$$

It is important that W_5 be higher than W_2 and W_3 or the interaction speed cannot be restored, but it should generally be set lower than W_1 or variations in speed become very unlikely.

In addition to these costs, another important point is that these candidate solutions may also lead to collision. Therefore, each of them must be tested in the same way as it has been explained for the initial trajectory. Feurtey simply discards every solution in which a potential collision is detected. Our approach is different: we process as an additional cost the danger represented by potential collisions, and argue that this cost may be more or less acceptable by different individuals. We also consider that this cost is proportional to the imminence of the collision. Given:

- the resulting cost of collision danger $C_6 \in \mathbf{R}$: $0.0 <= C_6 <= 95.0$,
- a weight parameter for the cost of collision danger $W_6 \in \mathbf{R}$: $0.0 <= W_6 <= 1.0$,
- the lowest frame number where a collision is detected $F \in \mathbf{N}_+$,
- the number of predicted frames $N \in \mathbf{N}_+$

$$C_6 = W_6 95 \frac{N - F}{N - 1}$$

This cost is potentially a lot higher that the others, since the mechanism must be able to operate like Feurtey's, i.e. an action is taken as soon as a future collision is detected, whatever its imminence. An advantage of this approach is that, when the density is very high and no collision-free solution is available, the characters will automatically select the candidate trajectory which has the highest probability of improving the situation.

The calculation of these costs is done every frame and not only when a collision is detected, since the characters' initial direction and speed must be restored once a collision has been avoided. These costs are used to compute a global score for the candidate trajectory, given:

- the resulting candidate trajectory's score $S \in \mathbf{R}$: $0.0 <= S <= 100.0$

$$S = 100 - K_1 - K_2 - K_3 - K_4 - K_5 - K_6$$

The candidate trajectory with the highest score is selected.

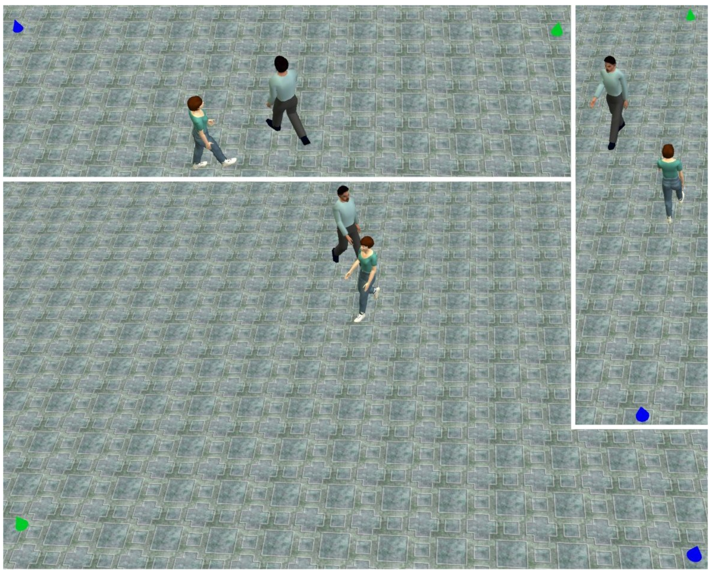


Figure 69 Different avoidance strategies in different situations: (from the upper left to the lower right) acceleration, direction change and deceleration

6.5.4.2 Social Characterization

The social characterization of the collision avoidance mechanism is made on two levels: (1) A collision is defined as a violation of personal space (whose features are dependent on the participant's social relationship) rather than physical contact - as described in the previous section (2) The costs of candidate trajectories are calculated differently for every character, in order to represent their individual preferences and social identity.

The main theoretical argument on which we base our approach is the following: it has been demonstrated that there exists, in small groups, a negative correlation between having power/prestige and the display of physical activity. More specifically, displacement of the body mass and high energy expenditure in the presence of others are signs of low status and/or deference (Schwartz, Tesser & Powell 1982). This applies perfectly to collision avoidance behaviours.

On the basis of this conception and of a number of well known social norms, like the "women first" norm, we have written rules for the calculation of the W_1-W_6 weight parameters. We do not provide the details of the calculation which is straightforward, but describe the SIML criteria we have selected and their influence:

1. **Roles and status**: Women are more passive than men in collision avoidance situations (lower W_6), and have relative high costs in direction change (W_1) and deceleration (W_3); high power and prestige members are generally passive (low W_6), but when they really have to avoid collision, use acceleration (low W_2).
2. **Individual preferences based on traits**: Children and adolescents have relative high costs in deceleration (W_3), but children easily accelerate (low W_2) and change direction (low W_1); on the contrary, the favoured avoidance strategy used by elders is slowing down (low W_3 but high W_2); in a similar way, extraverts prefer accelerating and changing direction (low W_2 and W_1), while introverts often choose to slow down (low W_3).
3. **Interaction**: Group members involved in an interaction with very low or very high Immediacy levels do not feel very much concerned about collision avoidance (low W_6) and, in any case, try to maintain a walking speed appropriate for the interaction (high W_5); low Relaxation interactions favour direction changes and acceleration (low W_1 and W_2) while high Relaxation interactions enforce the strategy of deceleration (low W_3).

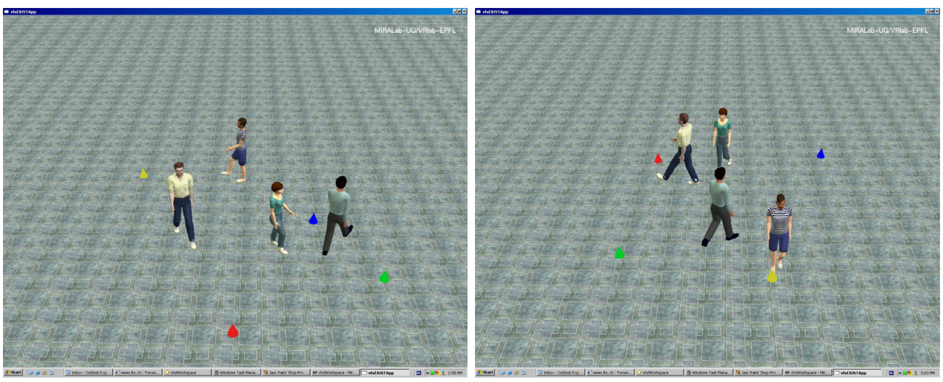


Figure 70 Social collision avoidance in a small group

Typically, when people with different genders and social status walk on conflicting trajectories, they can adopt one of their preferred ways of avoiding collision (accelerating, slowing down or changing direction). On the contrary, when two same-gendered high status members walk toward one another on a orthogonal trajectory, they wait to be close to each other before taking action. In the case of two particularly high status members or if several passivity criteria enforce each other, it can even happen that no one wishes to make the effort of avoiding the other and that collision occurs. However, this cannot be considered a failure of our mechanism since collision is one of the possible scenarios of real-life collision avoidance situations.

6.6 Simulation Of Kinesics

The procedural postural shifts, adaptor gestures and observation movements that we have presented are examples of kinesic activity. However, many other types of movements are involved in nonverbal social interaction: for example, the actions of the lips and of the arms that accompany speech (identified as requirements in Chapter 2), head nods, appreciative or depreciative gestures of the hands, etc. For all these actions, we can technically rely on predefined face and body animations, but the main issue is to decide which animations to trigger and when.

The Immediacy, Relaxation and Responsiveness dimensions should logically play a central part in this choice. However, in order to link a number of predefined animations to a specific interaction, the characterization of the interaction on these dimensions is not sufficient: a way of providing the position of the predefined animations themselves in the three dimensional framework is required. In addition, it should be noted that Mehrabian's framework successfully accounts for the form of nonverbal acts, but does not include information about the functional requirement they fulfil or their place in the interaction structure. This type of information is extremely important to decide if an animation should be triggered by the initiator or the receiver(s), at the beginning or at the end of the interaction, etc. Consequently, we propose an XML language for the characterization of synthetic social and emotional displays, that incorporates these different dimensions.

6.6.1 Social and Emotional Displays Markup Language

The Social and Emotional Displays Markup Language (SEDML) allows for the coding of predefined animations involved in social interaction. When large databases of animations are built, it is not appropriate that developers have to visualize every sequence in order to decide which to use or to rely completely on *ad hoc* descriptions. SEDML offers a convenient way of creating comprehensive and formal descriptions of animations that can be later used by human users or processed by automated systems.

Figure 71 shows the SEDML syntax. <*SEDisplay*> is the top-level tag. It describes a face or body action that is of general use or specific to a group (in which case the *GroupId* attribute is set). The first field contains the type of action; the available choices are adapted from Ekman and Friesen's (1969) categories: Adaptor (object or self manipulations, previously described), Deictic (actions that are used to identify an object or a class of objects, e.g. pointing gestures), Emblem (gestures having a precise meaning that can be translated by one or two words, like head nods meaning "yes" or thumb up for "good"), Facial expression (a type of affect display), Facial illustrator (facial actions used with speech, e.g. lips movements), Illustrator (gestures that complement verbal meaning, e.g. sweeping gestures of the arms), Postural shift (another type of affect display, previously described), Turn signal (a type of regulatory gesture used to signal that one is about to talk[11]).

Additional information must be provided on the same layer: the action's name, a reference to the corresponding animation file (in our implementation, .fap and .wrk files), the animation's duration in seconds, a textual description, an evaluation of the action's Immediacy, Relaxation and Responsiveness levels, and a frequency parameter specifying if the usage of the action is common or rare.

```
<SEDML xmlns:SIML="http://vrlab.epfl.ch/~aguye/SEDML">
<SEDisplay GroupId="integer 1 <= i" - optional >
  <Type>string Adaptor,Deictic,Emblem,FacialExpression,FacialIllustrator,
  Illustrator,PosturalShift,TurnSignal</Type>
  <Name>string "name of display"</Name>
  <File>string "path and filename"</File>
  <Duration>float 0.0 <= f</Duration>
  <Description>string</Description>
  <Immediacy>float 0.0 <= f <= 1.0</Immediacy>
  <Relaxation>float 0.0 <= f <= 1.0</Relaxation>
  <Responsiveness>float 0.0 <= f <= 1.0</Responsiveness>
  <Frequency>float 0.0 <= f <= 1.0</Frequency>
  <UseContext - optional>
    <InteractionPhase>string Contact,Midterm,Ending</InteractionPhase>
    <InteractionRole>string Initiator,Receiver</InteractionRole>
    <DisplayAgent - multiple>
      <Gender>string Female,Male</Gender>
      <AgeGroup>string Child,Adolescent,Adult,Elder</AgeGroup>
      <Culture>string WesternEurope,Russia/EasternEurope,NorthAmerica,
       MiddleAmerica,SouthAmerica,NorthAfrica/SouthwestAsia,SubSaharanAfrica,
      EastAsia,SoutheastAsia,Australia,PacificRealm</Culture>
    </DisplayAgent>
    <DestinationAgent - multiple>
      <Gender>string Female,Male</Gender>
      <AgeGroup>string Child,Adolescent,Adult,Elder</AgeGroup>
      <Culture>string WesternEurope,Russia/EasternEurope,NorthAmerica,
       MiddleAmerica,SouthAmerica,NorthAfrica/SouthwestAsia,SubSaharanAfrica,
      EastAsia,SoutheastAsia,Australia,PacificRealm</Culture>
    </DestinationAgent>
    <Interaction - multiple >
      <Type MinSuccess="float 0.0 <= f <= 1.0" MaxSuccess="float 0.0 <= f <= 1.0">string
       ShowSolidarity,ShowTensionRelease,Agree,GiveSuggestion,GiveOpinion,GiveOrientation,
      AskForOrientation,AskForOpinion,AskForSuggestion,Disagree,ShowTension,ShowAntagonism
      </Type - multiple>
    </Interaction>
  </UseContext>
</SEDisplay>
</SEDML>
```

Figure 71 The SEDML syntax

[11] Duncan and Fiske (1977) describe very precisely the nonverbal cues used as turn signals and within-turn signals, as well as the different back channel actions (head nods and shakes, gesticulations, etc.).

The second part of the format defines precisely the different use cases for the action. The *<UseContext>* tag is however optional, and is only present when it is necessary to provide detailed specifications. When an information is not provided, it means that no use restrictions are required on this aspect.

The first information that can be provided is the interaction phase in which the action may be triggered: typically, greeting gestures will appear in the Contact or Ending phases. Similarly, it can be specified if the usage of the action must be limited to the interaction's initiator or receiver; illustrators are generally assigned to the initiator and acknowledging gestures such as head nods to the receiver. The *<DisplayAgent>* tag makes use of SIML data to describe the required profile of the performer of the action. Several different profiles can be provided, as it is the case for the profiles of the destination agents. When *<DestinationAgent>* tags are present, it means that the action can be selected by an agent only when it is interacting with another agent corresponding to one of the profiles. Finally, precise information about the interaction in which the action is to be used can be provided in the *<Interaction>* tag. The triggering condition is that the ongoing interaction be of one of the specified types and that its success score match the provided range. For example, an animation corresponding to a particularly good laugh could be coded using "*<Type MinSuccess="0.8" MaxSuccess="1.0 ">ShowTensionRelease</Type>*".

Our SEDML parser does not limit the number of *<DisplayAgent>*, *<DestinationAgent>*, and *<Interaction>* sections provided. However, the SEDML data is pre-processed in order to simplify and accelerate the operation that tests whether a display matches the interactional context, e.g. the agent's characteristics are arranged in a multi-dimensional array of booleans.

6.6.2 The Equilibrium Theory

On the basis of the interaction's scores in Immediacy, Relaxation and Responsiveness and of SEDML data characterizing a number of social and emotional displays, different models can be proposed for the triggering of animations. Our approach is inspired by Argyle and Dean’s (1965) Equilibrium Theory.

The Equilibrium Theory states that individuals engaged in face-to-face interactions try to establish, through a number of nonverbal actions, a specific level of intimacy. These actions can be gestures, postural shifts, eye contact, distance choices, etc. According to Argyle and Dean, nonverbal actions are governed by two types of forces: (1) approach forces, which are related to the need for affiliation and (2) avoidance forces, which originate in the fear of rejection. Once an equilibrium point between these forces is reached in an interaction, it means that an appropriate level of intimacy has been established. Participants then try to maintain this level of intimacy by triggering compensatory actions when it is required. These compensatory actions counteract imbalances due to previous actions that took place using the same or a different channel (e.g. body parts), or to environmental constraints. A classic example describes two people interacting in an elevator; when another person enters and the participants must adopt a very close position, the rate of eye contacts decreases and they naturally choose less personal topics of conversation in order to maintain the overall intimacy level.

Adapted to our approach, the model describes how participants establish and maintain the interaction levels of Immediacy, Relaxation and Responsiveness as close as possible to the theoretical scores, by selecting adequate nonverbal actions on the basis of SEDML data. Since our parametric behaviours (e.g. postures) are designed to perfectly match the interaction scores and since we do not model environmental constraints, no imbalances are produced in the work previously presented. However, predefined animations rarely correspond exactly to a precise interactional context. Frequently, an animation matches one dimension but not another, thus correcting an imbalance but generating a new one. We take advantage of this characteristic: our work is concerned with the generation of compensatory actions of the kinesic type counterbalancing other kinesic actions. The Kinesics model we present aims at minimizing the overall distance between the real and ideal interaction scores through oscillatory fluctuations in Mehrabian's three dimensions, and at generating lively interactions made up of a variety of predefined animations.

6.6.3 A Kinesics Model Based on the Equilibrium Theory

Our Kinesics model is composed of two main parts: (1) the specification of how the triggering of displays influences the state of a nonverbal interaction (2) a mechanism for the selection of appropriate displays given the state of a nonverbal interaction.

6.6.3.1 Nonverbal Interaction Evolution

When a nonverbal interaction is initiated, it is assigned a neutral state. Our definition of an interaction's neutral state is the following: Immediacy 0.5, Relaxation 0.5, Responsiveness 0.0. In this section, we explain how these initial values are modified by the triggering of predefined animations.

As it appears in the selected values, Responsiveness is, in our approach, of a somewhat different nature than Immediacy and Relaxation. This is because it is not only related to the form of the triggered animations, but also to the number of animations that have been triggered. Typically, high Responsiveness levels in an interaction can be reached using a few high Responsiveness displays, as well as with many low Responsiveness displays. This "additive" nature of Responsiveness explains why our calculation of Responsiveness updates is not identical to Relaxation and Immediacy updates. For the latter, we compute an average value taking into account the effect of the new and of the previously triggered animations. Given:

- the resulting interaction Immediacy score $I' \in \mathbf{R}$: $0.0 <= I' <= 1.0$,
- the previous interaction Immediacy score $I \in \mathbf{R}$: $0.0 <= I <= 1.0$,
- the triggered display's Immediacy score $I_d \in \mathbf{R}$: $0.0 <= I_d <= 1.0$,
- a global parameter representing the participants' short-term memory for previous displays which determines the speed of evolution of the interaction $M \in \mathbf{N}_+$

$$I' = \frac{MI + I_d}{M + 1}$$

Some nonverbal actions do not exist in the Immediacy dimension, like adaptors which have no affiliation connotation. Since using a 0.5 value would artificially alter the interaction, the corresponding SEDML field is left empty. In this case, the Immediacy score is simply not updated.

Here is the calculation of Relaxation's updates, which are similar to Immediacy's. Given:

- the resulting interaction Relaxation score $R' \in \mathbf{R}$: $0.0 <= R' <= 1.0$,
- the previous interaction Relaxation score $R \in \mathbf{R}$: $0.0 <= R <= 1.0$,
- the triggered display's Relaxation score $R_d \in \mathbf{R}$: $0.0 <= R_d <= 1.0$

$$R' = \frac{MR + R_d}{M + 1}$$

Contrary to Immediacy and Relaxation updates, the triggering of a display always increases the interaction's Responsiveness score. Given:

- the resulting interaction Responsiveness score $O' \in \mathbf{R}$: $0.0 <= O' <= 1.0$,
- the previous interaction Responsiveness score $O \in \mathbf{R}$: $0.0 <= O <= 1.0$,
- the triggered display's Responsiveness score $O_d \in \mathbf{R}$: $0.0 <= O_d <= 1.0$,
- a global parameter determining the relative speed of Responsiveness development $S \in \mathbf{R}_+$

$$O' = O + \frac{O_d}{MS}$$

In the absence of any activity, the level of interaction Responsiveness must automatically decrease, and, for this purpose, we introduce a Responsiveness decay function which is executed every frame. We take into account the fact that once the target Responsiveness score has been reached, the rate of animation triggering must still be higher in a high than in a low Responsiveness interaction. Therefore, the decay speed is proportional to the target Responsiveness score. Given:

- the target interaction Responsiveness score $O_t \in \mathbf{R}$: $0.0 <= O_t <= 1.0$,
- a global parameter determining the speed of Responsiveness decay $D \in \mathbf{R}$: $0.0 <= D <= 1.0$

$$O' = O - DO_t$$

In order to make the evolution model more realistic, and because an action may be interrupted before it is completed, the difference between the new and previous scores is not entirely added/subtracted at the beginning or the end of the animation. Rather, a small part is added/subtracted every frame until the animation is over. We consider that the temporal distribution of meaning in a gesture or facial expression can be approximated using a Gaussian distribution. We base our choice on the fact that many nonverbal actions comprise an initial and a terminal phase in which a body part is moved to/from a given location before/after that the truly meaningful action takes place. Using this approach, if an animation is interrupted shortly after that it has been triggered, the impact on the interaction will be minimal.

6.6.3.2 Display Selection Mechanism

Every frame, it must be determined if one of the available animations is suitable to the ongoing interaction, given the current Immediacy, Relaxation and Responsiveness levels. The following algorithm is used when a kinesic action is not already being executed:

1. If not previously done, the system configuration .sedml file is loaded and parsed.
2. All SEDML displays which do not match the ongoing interaction's context are automatically filtered out.
3. Additional filtering can take place according to the type of display and to the interaction structure; e.g. in Self interactions, only adaptors are considered.
4. All displays having a score in Immediacy, Relaxation or Responsiveness whose difference with the corresponding target score surpasses a given threshold are also filtered out.
5. A general suitability score is calculated for all remaining displays.
6. The animation corresponding to the display having the highest suitability score is triggered if this score also surpasses that of a void display having no effect on the interaction.

This final test is useful in order to temporarily stop the triggering of animations when the target scores are reached, and to prevent the selection of displays having a negative impact on the interaction. As soon as the Responsiveness score has decreased to a significant extent because of the decay function, activity resumes.

Although an initial pass removes the grossly inadequate displays on the basis of each dimension separately, it is the suitability score which serves to minimize the 3D distance between the current and target interaction scores. In order to calculate this score, the triggering of the evaluated display is simulated and the resulting Immediacy, Relaxation and Responsiveness scores are predicted using the formulas described in the previous section. Given:

- the resulting suitability score $U_d \in \mathbf{R}$: $0.0 <= U_d <= 2.0$,
- the target interaction Immediacy score $I_t \in \mathbf{R}$: $0.0 <= I_t <= 1.0$,
- the target interaction Relaxation score $R_t \in \mathbf{R}$: $0.0 <= R_t <= 1.0$,
- the target interaction Responsiveness score $O_t \in \mathbf{R}$: $0.0 <= O_t <= 1.0$,
- the predicted interaction Immediacy score $I_p \in \mathbf{R}$: $0.0 <= I_p <= 1.0$,
- the predicted interaction Relaxation score $R_p \in \mathbf{R}$: $0.0 <= R_p <= 1.0$,
- the predicted interaction Responsiveness score $O_p \in \mathbf{R}$: $0.0 <= O_p <= 1.0$,
- the SEDML display frequency parameter $F_d \in \mathbf{R}$: $0.0 <= F_d <= 1.0$,
- a random value $D \in \mathbf{R}$: $0.0 <= D <= 1.0$

$$U_d = 1 - \sqrt{(I_t - I_p)^2 + (R_t - R_p)^2 + (O_t - O_p)^2} + DF_d$$

An example of the generated interaction profiles is given in Figure 72. The name of the triggered animations is included, and their effects on the three dimensions appear clearly. The automatic decay of Responsiveness is also visible, as well as the transitions computed using a Gaussian distribution function.

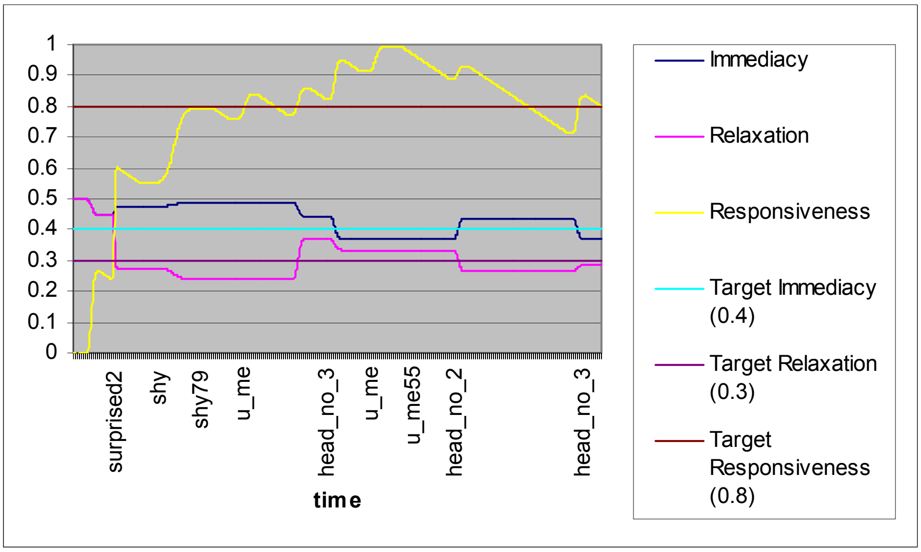


Figure 72 Example profile of kinesic activity in a nonverbal interaction

In order to decide when the interaction can be terminated, a variable is incremented every frame in which the 3D distance between the actual and target scores is lower than a given threshold. When this variable reaches a value depending on the application requirements, the interaction enters its Ending phase. By selecting a particularly low value for this threshold as well as high values for the parameters determining the evolution speed, the system can be configured to produce very short interactions triggered in bursts. In this case, only one or two nonverbal actions are sufficient to constitute an IPA interaction.

Figure 73 Simulation of kinesic activity in a dyadic interaction

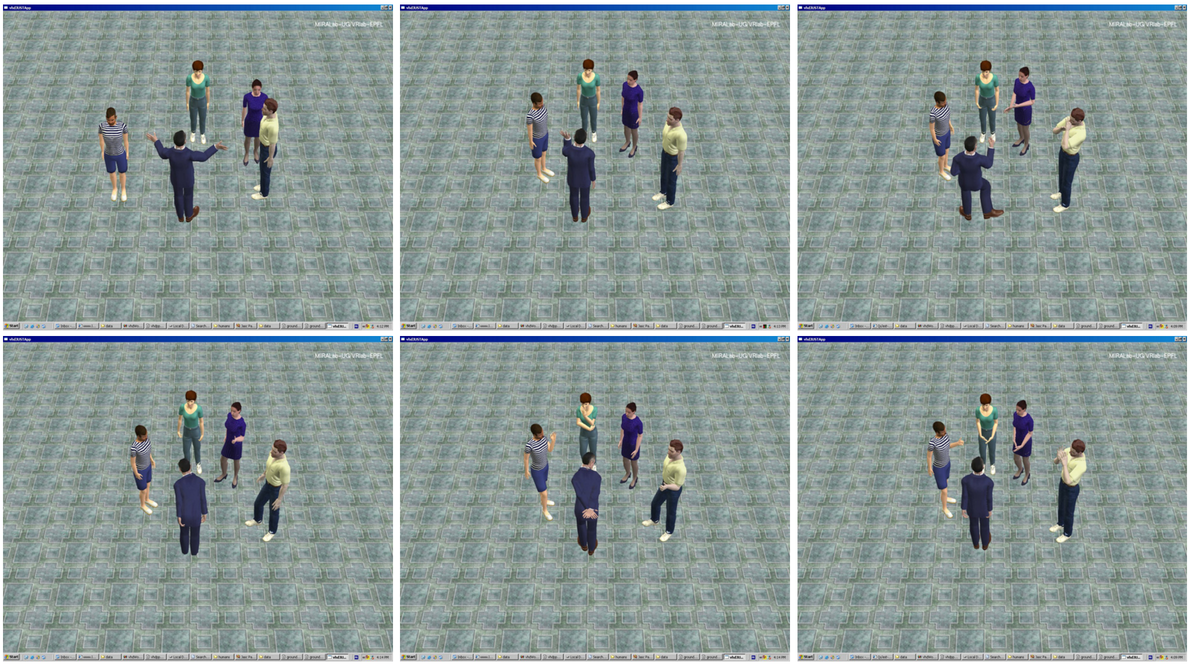


Figure 74 Simulation of kinesic activity in a group interaction

As it appeared in the previous descriptions, our approach to the simulation of Kinesics aims at producing a believable statistical approximation of the actions triggered in face-to-face encounters according to the nature of the participants' interaction. Although the actions we trigger are not explicitly linked to one another, the fact of taking into account their cumulative effects on the Immediacy, Relaxation and Responsiveness interaction scores contributes to the generation of meaningful sequences of kinesic actions. It must be underlined that the performance of our algorithm is directly dependent on the number and diversity of the predefined animations provided and described in the SEDML system configuration file.

6.7 Integration in VHD++

As previously mentioned, our work is integrated in the VHD++ development framework (Ponder et al. 2003). VHD++ supports developers in the rapid and component-based development of 3D simulation applications on PC, requiring the advanced visualization and animation of virtual characters. It is a strongly object oriented framework, entirely developed in C++, which comprises more than 500 classes grouped in 35 packages.

The software architecture is described in a paper by Ponder et al. (2003). It includes two main types of components: (1) *vhdRuntimeEngine*: a reusable high performance operational kernel which is in charge of initialising the system, loading configurations and multimedia content, data sharing, multi-threading and synchronization, event brokering, garbage collecting, etc. (2) *vhdServices*: these are plugable software components which control some specific aspects of the simulation and allow application developers to conveniently access various technologies like 3D rendering, 3D sound, skeleton animation, etc.

The work we have presented in this chapter is implemented as a high-level VHD++ service called *vhdNonVerbalSocialInteractionService*. It mainly makes use of two other services: *vhdHAGENTServiceBody* for the management of the characters in the 3D scene and body animation, and *vhdFaceAnimServiceBody* for facial animation. Figure 75 gives an overview of the integration in VHD++ and with the work presented in Chapter 5.

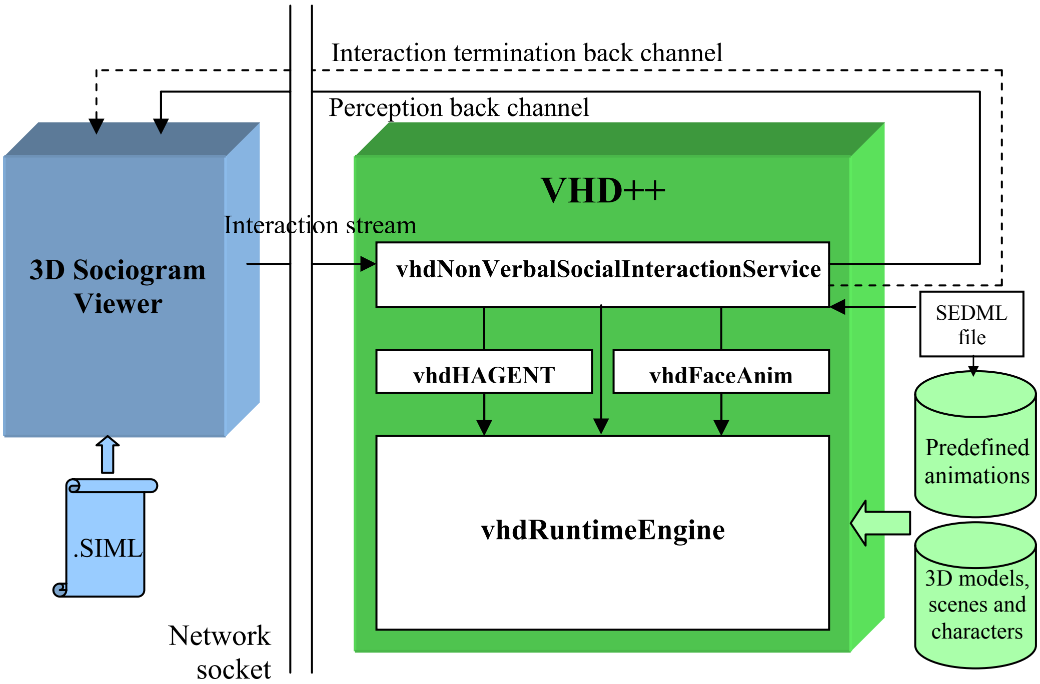


Figure 75 Integration in VHD++ and with 3DSociogramViewer

As it can be seen, 3DSociogramViewer and our VHD++ service communicate through a network socket. Every time an interaction is triggered in 3DSociogramViewer, a message describing the type of interaction, the interaction success, the initiator and receiver(s) IDs as well as the full interaction context, which comprises the participants' SIML profiles, is sent to VHD++. When the interaction's visualization is completed (i.e. kinesic simulation succeeded), 3DSociogramViewer is informed through a dedicated back channel. A second back channel is used to send, each frame, the list of perceived agent IDs for every character, in order to enforce their motivation to interact as described in Section 5.4.2.1. Although a network socket is in charge of the communication between the two applications, they can be run on the same system: 3DSociogramViewer is in this case used for the visualization of the internal group processes and for their control through manually triggered interactions, while the corresponding nonverbal behaviours can be monitored on the VHD++ display (see Figure 77). However, since VHD++ is a CPU intensive and memory demanding application, it is often more convenient to use different clients and servers for simulations involving complex 3D models.

Figure 76 represents the main classes used in *vhdNonVerbalSocialInteractionService.* The most characteristic attributes and methods of these classes have been included. It can be seen that the *SAgent* class occupies a central position in the architecture and is in charge of all the low-level tasks such as walking to a given location, avoiding collisions, facing, etc. It also serves as an interface to the *ParametricObservation*, *ParametricAdaptors* and *ParametricPosture* classes as well as to the *KinesicsEngine*. The three types of social interaction we have defined (*SelfInteraction, GroupInteraction, DyadicInteraction*) use *SInteraction* as a base class which contains their common data and methods. An object of the *vhdNonVerbalSocialInteractionServiceBody* class is automatically instantiated when VHD++

starts up, and is called every frame. It manages a list of *SAgent* and *SInteraction* references, is in charge of the communication tasks, and successively gives the control to every active *SInteraction* when it is required.

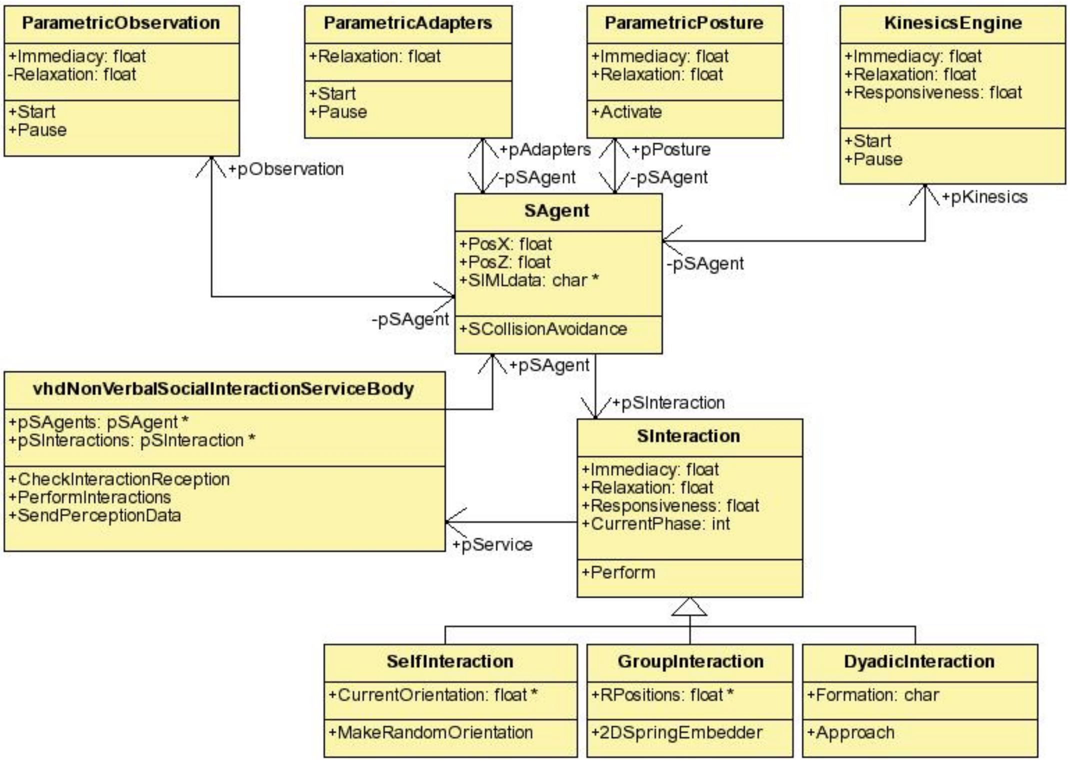


Figure 76 A UML diagram representing the main classes used in vhdNonVerbalSocialInteractionService

Other VHD++ services can easily use our interaction visualization system by creating suitable *SInteraction* objects and by adding them to the list of references handled in *vhdNonVerbalSocialInteractionServiceBody*.

6.8 A Small Group Interacting in VRLab's El Farol Bar

Bars and pubs are known to be among the most sociopetal public places, i.e. they bring people together and significantly encourage social interaction (Hall 1966; Mehrabian 1976). It was therefore natural to conclude this chapter with two examples of interaction taking place in such an environment.

In allusion to Brian Arthur's (1994) work, the bar in which our interactions take place is VRLab's El Farol's. Arthur's famous "El Farol Bar" problem consists in a number of agents having to decide whether to go to the bar on Thursday night or stay at home. If they anticipate that the bar is going to be crowded, they don't go; otherwise, they decide to have a drink there. Since the bar is going to be crowded when every agent anticipates that it will not, this problem illustrates the self-defeating properties of shared models in multi-agent systems. In our examples, we are concerned with the visualization of the nonverbal interactions of a few agents having decided to go to the bar together on a typically less-crowded Wednesday night.

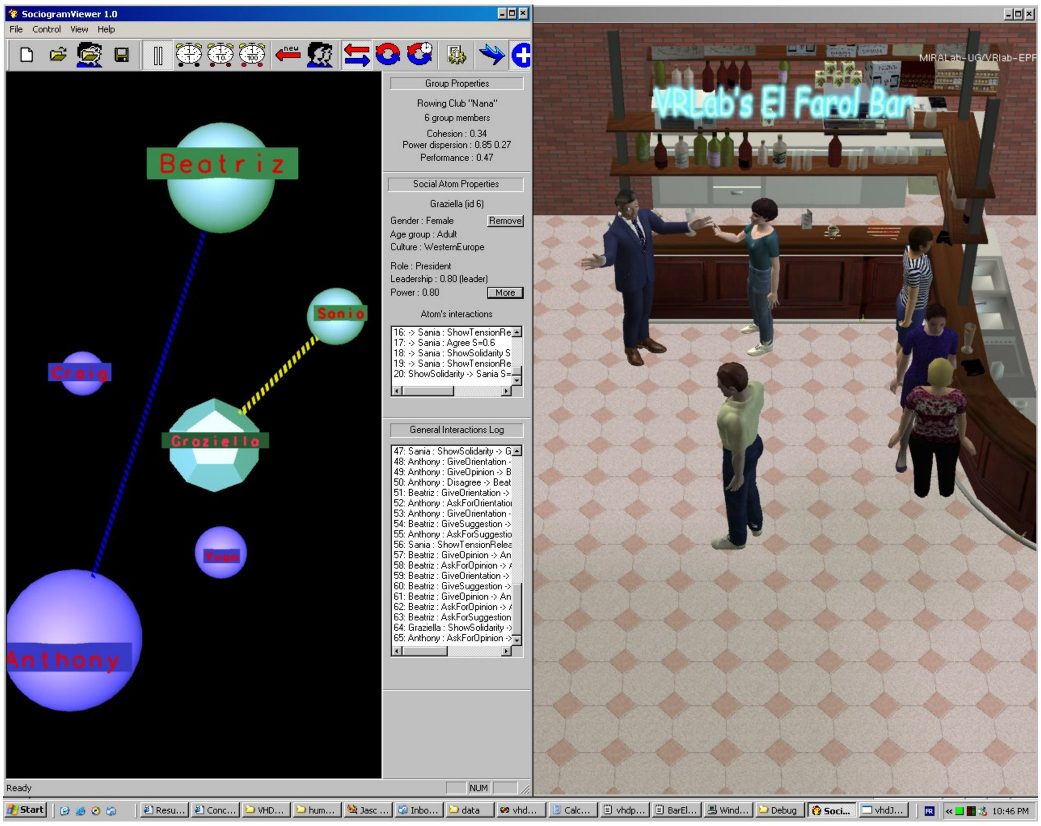


Figure 77 A small group and their dyadic interactions taking place in VRLab's El Farol bar, simultaneously represented in 3DSociogramViewer and VHD++

Figure 78 A low Immediacy group interaction in VRLab's ElFarol bar

6.9 Conclusion

In this final chapter, we have presented our approach of the visualization of nonverbal social interaction. On the basis of a theoretical framework in which Mehrabian's ideas play a central part, it has been explained how the proxemic aspects of nonverbal interactions can be simulated with a high level of details. For the simulation of Kinesics, a number of specialised motion synthesis functions, as well as a general mechanism triggering predefined animations, have been proposed. In all these contributions, we have tried to consistently use Immediacy, Relaxation and Responsiveness as the main determinants of nonverbal communication and nonverbal behaviour in general.

However, a number of important features need to be improved or are still missing in our simulation of Kinesics and Proxemics:

- Observation behaviours should comprise movements of the eyes and pupils.
- Touch gestures should be parametrically controllable (currently, we rely on predefined animations which cannot be accurately performed on characters varying in size and shape).
- In the parametric generation of postures and adaptors, more joints should be taken into account in order to reduce the impression of stiffness.
- Parametric illustrators would match more closely the ongoing speech.
- In addition to turn taking gestures, providing and coding other types of regulator gestures would enforce the simulation of the nonverbal interactions' structural dimension.
- Approach behaviours need to be improved, especially for group interactions.
- Our method for positioning and orienting the participants involved in group interaction should be validated.

A virtually endless list of improvements and additions could be drawn up because of the complexity of the simulated phenomenon. However, we think that the mentioned points should be addressed in priority.

The integration of our work in VHD++ went smoothly, and the choice to keep 3DSociogramViewer and the nonverbal interaction visualization service separate reduces the overall complexity of the system. It also has the advantage of allowing other developers, involved for example in the simulation of crowds, to exploit the visualization part of our work without having to manage the full simulation system. With the contributions described in this chapter, which complement the previously presented work, we feel that our investigation has come full circle.

Chapter 7 • Conclusion

7.1 Summary

In this thesis, we have investigated the problems and solutions for the simulation of nonverbal social interaction and group dynamics in virtual environments. After an initial analysis of the task requirements as identified in a small-scale evaluation experiment, we have presented a number of technological components based on MPEG-4 for the visualization of face and body activity, and discussed the specificity of social behaviour in terms of agent modelling and simulation. In order to generate meaningful sequences of social interactions in the temporal dimension, a model of action selection in the context of small groups dynamics along with a model of interpersonal relationship development have been proposed. Finally, we have explained how these interactions can be visualised in virtual environments using a dedicated VHD++ service.

More specifically, the main contributions made in the framework of this thesis are:

- an evaluation of the way people communicate nonverbally in virtual environments,
- reusable MPEG-4 software components for body animation,
- AML (syntax and parser implementation): a tag-based language for the seamless integration and synchronization of facial animation, body animation, and speech,
- a conceptual architecture for the modelling of autonomous social agents,
- SIML (syntax and parser implementation): a tag-based language for the modelling of agents' social identity and group belonging,
- a model of action selection in the context of small groups dynamics,
- a model of interpersonal relationship development in small groups,
- 3DSociogramViewer: a tool for the visualization and control of interpersonal relationships and interactions in small groups,
- motion synthesis components for the generation of parametric postures, adaptors and observation behaviours,
- a number of mechanisms for the simulation of Proxemics in dyadic and group interactions, including the social avoidance of collision,
- SEDML (syntax and parser implementation): a tag-based language for the characterization of nonverbal actions in the context of social interaction,
- a mechanism based on predefined animations for the simulation of kinesic activity during social interaction

Three types of nonverbal social interaction have been examined in the research: Human–Human (Chapter 2), Human–Agent (Chapter 3), and Agent–Agent (Chapter 5 and Chapter 6). However, the most important aspects of the work have been the adequate triggering of social interactions by autonomous agents, and the believable representation of the corresponding nonverbal behaviours using 3D characters. Interestingly, the way these tasks have been addressed can be conceived as a process of step-by-step transformation of a number of high-level parameters into low-level parameters. The initial high-level parameters take the form of SIML data, which describe the group members' social identity, and are used to trigger appropriate interactions. On the basis of the characteristics of the triggered interactions and the profiles of the participants, scores for Immediacy, Relaxation and Responsiveness are calculated. Finally, these intermediate parameters are transformed into joints angle values or features points displacements, using procedural animation or executing carefully selected predefined animations.

Through our work, we have also shown how inputs from social sciences can be used for the creation of multi-modal interactions and multi-agent environments, and identified many references that could prove useful for computer scientists involved in these research domains.

7.2 Limitations and Future Work

When tackling such a large and complex topic as the simulation of nonverbal social interaction, there are many limitations of the proposed solutions. The main limitations of this work are related to the greatly simplified model of the interaction context. In order to generate interactions and give them substance, we rely on a number of widespread social norms, roles and general factors which have been documented as influencing the interaction process. Such an approach has a high degree of generality; however, the specifics of each situation can never be completely captured. This limitation manifests itself at different levels:

- The social characteristics and personality of the involved group members are only approximated with SIML data; furthermore, the information which can already be provided is certainly not fully exploited.
- Twelve types of interactions can only grossly account for real-life interactions; as an example of this limitation, Bales himself had to include in his descriptions a variety of nonverbal behaviours for the same IPA type.
- The group's social environment is not modelled; for instance, inter-group interactions should be taken into account in addition to intra-group interactions.
- Even "social" groups generally have a specific purpose which influences their members' behaviour[12]; in our work, no performance requirements or resources limitations are involved.
- The physical environment in which interactions occur is not taken into account in the participants' nonverbal behaviour; in real-life, people are constrained by, but can also instrumentalize, obstacles and other characteristics of the place of interaction.

These limitations should be considered as advanced requirements to be addressed in future research. However, it must be underlined that a trade-off exists between the generality of the provided contributions and their ability to account accurately for specific interactional contexts. When the goal is to explore and simulate a variety of interactions, our approach is perfectly appropriate. In order to reproduce realistically very precise situations in 3D,

[12] This is at least the case for "pour-soi" (being-for-itself) groups, in contrast with "en-soi" (being-in-itself) groups composed of people who are not aware of being part of a common entity.

however, it may be necessary to complement it by means of a manual design, and *ad hoc* choices.

In addition to the improvements related to the modelling of the interaction context, the simulation itself may be extended as follows:

- Knapp's relationship termination model could be integrated in addition to relationship development; similarly, the voluntary exit and the exclusion of group members could be automated.
- The initial formation and termination of small groups, on the basis of a population of predefined agents, could be modelled.
- The assumption that group members fulfil their assigned social roles, follow norms and act in the interest of their group, is not always true; by merging *Homo economicus* and *Homo sociologicus* as suggested in Chapter 4, human behaviour would be simulated more completely.
- The formation and learning of roles and interaction norms could be modelled: for example, when a new member enters a group, he/she could learn through trial and error the appropriate interaction rates for every other member; in this case, the output of the fitness function could be the already included interaction success score.
- As described in the preceding chapter, the simulation of kinesics should be extended by integrating new actions such as eye movements.
- The simulation of Proxemics should also be extended by managing interaction types other than standing interactions: for example, seating choices motivated by the available arrangements and the participants' social relationships have been widely studied and could be easily reproduced.
- Additional nonverbal communication channels could be integrated: para-language (tone and pitch of voice), olfactory signals, visual outlook (clothes, hair style, artefacts), etc.; for example, the appearance of 3D characters, including suitable clothes, could be automatically generated according to the agent identities provided.

With the addition of such features, and the rapid improvement of 3D visualization through continuously increasing hardware performance, the degree of realism of the visual aspect of simulations may reach high levels in the near future. However, it is probable that the simulation of human reasoning will remain an issue for a longer period of time. Gödel (1965) has shown, with his "Incompleteness Theorem", that rigidly logical systems such as Turing machines may be confronted with undecidable propositions and, thus, that "there exist some tasks the mind can solve but machines cannot". In this context, the objective of research should certainly be to approach this limit, before a different paradigm emerges and opens new horizons for the simulation of human behaviour. The recent surge of interest in the simulation of social behaviour and social reasoning may well accelerate this process.

References

André, E., Müller, J. and Rist, T. (1996) 'The PPP persona: A multipurpose animated presentation agent', In: Catarci, T. et al. *Advanced Visual Interfaces: Proceedings of the International Workshop AVI'96, May 27-29, 1992, Rome, Italy*. World Scientific Press, Singapore. pp. 245-247.

Arafa, Y., Kamyab, K., Mamdani, E., Kshirsagar, S., Magnenat-Thalmann, N., Guye-Vuillème, A. and Thalmann, D. (2002) 'Two approaches to scripting character animation', In: *Proceedings of Embodied Conversational Agents - Let's Specify and Evaluate Them! at AAMAS 2002, Bologna, Italy*. New York: ACM Press.

Arafa, Y. and Mamdani, E. (2003) 'Scripting embodied agents behaviour with CML: Character Markup Language', In: *Proceedings of the International Conference on Intelligent User Interfaces, Miami, Florida, January 2003*. ACM Press, New York. pp. 313-316

Archer, M. (2000), Homo economicus, Homo sociologicus and Homo sentiens, in: Archer, M. and Tritter, J. (eds) *Rational Choice Theory: Resisting Colonisation*, Routledge, London & New York

Argyle, M. and Dean, J. (1965), "Eye contact, distance and affiliation", *Sociometry*, no. 28, pp. 289-304

Argyle, M. (1988), *Bodily Communication*, Methuen & Co., New York

Arthur, W.B. (1994), "Inductive reasoning and bounded rationality", *American Economic Review*, vol. 84, no. 2, pp. 406-411

Asimov, I. (1968), *I, Robot*, Grafton Books, London

Axelrod, R. (1981), "The Emergence of cooperation among egoists", *The American Political Science Review*, vol.75, pp. 306-318

Bales, R.F. (1951), *Interaction Process Analysis: A Method for the Study of Small Groups*, Addison-Wesley Press, Inc., Cambridge, MA

Bales, R.F. (1970), *Personality and Interpersonal Behavior*, Holt, Rinehart and Winston Inc., New York

Bales, R. F. (1980), *SYMLOG: Case study kit with instructions for a group self study*, Free Press, New York

Balzer, W. (2000), Smass: A sequential multi-agent system for social simulation, in: Suleiman, R., Troitzsch, K. and Gilbert, N. (eds) *Tools and Techniques for Social Simulation*, Physica-Verlag, Heidelberg

Bates, J., Loyall, A.B, and Reilly, W.S. (1994) 'An architecture for action, emotion, and social behavior', In: Castelfranchi, C. and Wemer, E. (eds) *Artificial Social Systems -- Selected Papers from the Fourth European Workshop on Modelling Autonomous Agents in a Multi-Agent World.* Springer, Heidelberg. pp. 55-68

Becheiraz, P. and Thalmann, D. (1998) 'A behavioral animation system for autonomous actors personified by emotions', In: *Proceedings of the First Workshop on Embodied Conversational Characters (WECC '98), Lake Tahoe, California*. pp. 57-65

Bedau, M., McCaskill, J. , Packard, N., Rasmussen, S., Adami, C., Green, D., Ikegami, T., Kaneko, K. and Ray T. (2000), "Open problems in artificial life", *Artificial Life*, vol. 6, pp. 363-376

Berger, J.O. (1985), *Statistical decision theory and Bayesian analysis*, Springer, New York

Berger, P.L. and Luckmann, T. (1966), *The social construction of reality: a treatise in the sociology of knowledge*. Anchor Books, Garden City, NY

Blaxxun Interactive homepage, (n.d.). <http://www.blaxxun.com/> (accessed 1st September 2003)

Blumer, H. (1969), *Symbolic Interactionism: Perspective and Method,* Prentice-Hall, Englewood Cliffs, NJ

Boman, M. and Verhagen, H. (1998), 'Social Intelligence as norm adaptation', In: Edmonds, B. and Dautenhahn, K. (eds) *Socially Situated Intelligence: a Workshop held at SAB'98, August 1998, Zürich.* University of Zürich, Zürich. pp. 17-24

Bordeux, C., Boulic, R. and Thalmann. D. (1999), "An efficient and flexible perception pipeline for autonomous agents", *Computer Graphics Forum*, vol. 18, no. 3, pp. 23–30

Botelho, L.M. and Coelho, H. (1998) 'Artificial autonomous agents with artificial emotions', In: Sycara, K.P. and Wooldridge, M. (eds) Proceedings of the Second International Conference on Autonomous Agents, St. Paul, Minnepolis, USA, May 9-13, 1998. ACM Press, New York

Boudon, R. (1979), *La Logique du Social*, Hachette, Paris

Boulic, R. et al. (1995) 'The HUMANOID environment for interactive animation of multiple deformable human characters', In: Post F.H. and Göbel M. (eds) *Proceedings of Eurographics '95, Maastricht, The Netherlands*. pp. 337-348

Bourdieu, P. (1990), *In other words: Essays Towards a Reflexive Sociology*, Stanford University Press, Stanford, CA

Bouvier, E. and Guilloteau, P. (1996) 'Crowd simulation in immersive space management', In: Göbel M. et al. (eds) *Virtual Environments and Scientific Visualization '96: Proceedings of the third eurographics workshop on virtual environments*. Springer, Wien. pp. 104–110

Bowers, J., Pycock, J. and O'Brien, J. (1996) 'Talk and embodiment in collaborative virtual environments', In: *Proceedings of the 1996 conference on Human Factors in Computing Systems -- CHI'96.* ACM Press, New York. pp. 58-65

Bratman, M. (1987), *Intentions, Plans, and Practical Reason*, Harvard University Press, Cambridge

Brooks, R.A. (1991), "Intelligence without representation", *Artificial Intelligence*, vol. 47, no. 1-3, pp. 139–159

Burgoon, J.K., Buller, D.B. and Woodall, W.G. (1996), *Nonverbal Communication: The Unspoken dialogue*, 2nd ed, McGraw-Hill Companies Inc., New York

Burke, P.J. (2003), Interaction in small groups, in: DeLamater, J. (ed) *Handbook of Social Psychology*, Kluwer Academic/Plenum, New York

Burt, R.S. (2000), "Decay functions, *Social Networks*, vol. 22, pp. 1-28

Caicedo, A. and Thalmann, D. (1999) 'Intelligent decision-making for virtual humanoids', In: Floreano D. et al. (eds) *Advances in Artificial Life: Proceedings of 5th European Conference ECAL'99*. Springer, Heidelberg. pp. 13-17

Capin, T.K. et al. (1997), "Virtual human representation and communication in VLNET networked virtual environment", *IEEE Computer Graphics and Applications*, vol. 17, no. 2, pp. 42-53

Capin, T.K. (1998), *Virtual Human Representation in Networked Virtual Environments*, thèse de doctorat, EPFL, Lausanne

Carley, K.M. (1991), "A theory of group stability", *American Sociological Review*, vol. 56, pp. 331-354

Carley, K.M and Newell, A. (1994), "The nature of the social agent", *Journal of Mathematical Sociology*, vol. 19, no. 4, pp. 221-262

Cass, S. (2002), "Mind games: to beat the competition, video games are getting smarter", *IEEE Spectrum Online*, Dec 02. <http://www.spectrum.ieee.org/WEBONLY/ publicfeature/ dec02/ mind.html> (accessed 1st September 2003)

Cassell, J., Vilhjalmsson, H. and Bickmore, T. (2001) 'BEAT: the Behaviour Expression Animation Toolkit', In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM Press, New York. pp. 477-486

Castelfranchi, C., Falcone, R., Firozabadi, B. and Tan, Y. (eds) (2000), *Applied Artificial Intelligence, Special Issue on "Trust in Agents" Part1*, vol. 14, no. 8

Castelfranchi, C., Falcone, R., Firozabadi, B. and Tan, Y. (eds) (2000), *Applied Artificial Intelligence, Special Issue on "Trust in Agents" Part2*, vol. 14, no. 9

Channon, A.D. and Damper, R.I. (1998) 'The evolutionary emergence of socially intelligent agents', In: Edmonds, B. and Dautenhahn, K. (eds) *Socially Situated Intelligence: a Workshop held at SAB'98, August 1998, Zürich.* University of Zürich, Zürich. pp. 41-49

Charlton, P. and Kamyab, K. (2000) 'Evaluating explicit models of affective interactions', In: Jeff Rickel (ed), *Proceedings of the workshop on "Achieving Human-Like Behavior in Interactive Animated Agents"*, held in conjunction with the Fourth International Conference on Autonomous Agents, Barcelona, 2000.

Cicourel, A.V. (1973), *Cognitive Sociology: Language and Meaning in Social Interaction*, Penguin, Harmondsworth

Ciolek, M.T. and Kendon, A. (1980), "Environment and the spatial arrangements of conversational encounters", *Sociological Enquiry*, vol. 50, pp. 237-271

Conte, R. and Castelfranchi, C. (1995), *Cognitive and Social Action,* UCL Press, London, UK

Conte, R., Castelfranchi, C. and Dignum, F. (1999), "Autonomous norm acceptance", *Intelligent Agents* V, no.1555, pp. 99-112

Corraze, G. (1980), *Les Communications Nonverbales*, Presses Universitaires de France, Paris

Crovitz, H.F. and Schiffman, H. (1974), "Frequency of episodic memories as a function of their age", *Bulletin of Psychonomic Society*, vol. 4, pp. 517-518

Dautenhahn, K. (1998), "The art of designing socially intelligent agents: science, fiction, and the human in the loop", *Applied artificial intelligence*, vol. 12, no. 7-8, pp. 573-617

Dautenhahn, K. (1999), Embodiment and interaction in socially intelligent life-like agents, in: Nehaniv C.L. (ed) *Computation for Metaphors, Analogy and Agents*, Springer Lecture Notes in Artificial Intelligence, vol. 1562

De Carolis, B., Carofiglio, V., Bilvi, M. and Pelachaud, C. (2002) 'APML, a mark-up language for believable behaviour generation', In: *Proceedings of Embodied Conversational Agents - Let's Specify and Evaluate Them! at AAMAS 2002*, *Bologna, Italy*. New York: ACM Press.

de Sevin, E., Kallmann, M. and Thalmann, D. (2001) 'Towards real time virtual human life simulations', In: Ip, H.S.H. et al. (eds) *Proceedings of the Computer Graphics International 2001 (CGI 2001)*. IEEE Computer Society Press, Washington. pp. 31-37

Dehn, D. and van Mulken, S. (2000), "The impact of animated interface agents: a review of empirical research", *International Journal of Human Computer Studies*, vol. 52, no. 1, pp. 1-22

Digman, J.M. (1990), "Personality structure: Emergence of the five-factor model", *Annual Review of Psychology*, vol. 41, pp. 417-440

Dignum, F., Morley, D., Sonenberg, E. and Cavedon, L. (2000) 'Towards socially sophisticated BDI agents', In: *Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS 2000)*. IEEE Computer Society Press, Washington. pp. 111-118

DOM homepage, <http://www.w3.org/DOM/> (accessed 1st September 2003)

Duncan, S.D. and Fiske, D.W. (1977), *Face-to-Face Interaction: Research, Methods, and Theory*. Erlbaum, Hillsdale, NJ

Durkheim, E. (1895), *Les Règles de la Méthode Sociologique*, Felix Alcan, Paris

Ekman, P. and Friesen, W.V. (1967), "Head and body cues in the judgement of emotion: a reformulation", *Perceptual Motor Skills*, vol. 24, pp. 711-724

Ekman, P. and Friesen, W.V. (1969), "The repertoire of nonverbal behavior: categories, origins, usage and coding", *Semiotica*, vol. 1, pp. 49-97

Feurtey, F. (2000), *Simulating the Collision Avoidance Behavior of Pedestrians*, Master's thesis, University of Tokyo, School of Engineering

Feyerabend, P.K. (1962), Explanation, reduction and empiricism, in: Feigl, H. and Maxwell, G. (eds) *Scientific Explanation, Space and Time*, University of Minnesota Press, Minneapolis

FIPA 97 homepage, 2002. <http://www.fipa.org/repository/fipa97.html> (accessed 1st September 2003)

Forsyth, D.R. (1999), *Group Dynamics*, 3rd ed, Wadsworth Publishing Co., Belmont, CA

Freeman, L. (2000), "Visualizing social networks", *Journal of Social Structure*, vol. 1, no. 1

Friedman-Hill, E. (2003), Jess, The Rule Engine for the Java Platform. *Distributed Computing Systems*. Sandia National Laboratories, Livermore, CA, Version 6.1p5. <http://herzberg.ca.sandia.gov/jess> (accessed 15th September 2003)

Garfinkel, H. (1967), *Studies in Ethnomethodology*, Prentice-Hal, Englewood Cliffs, NJ

Georgeff, M., Pell, B., Pollack, M., Tambe, M. and Wooldridge, M. (1999) 'The Belief-Desire-Intention model of agency', In: Müller, J., Singh M.P. and Rao A.S. (eds) *Proceedings of the 5th International Workshop on Intelligent Agents V: Agent Theories, Architectures, and Languages (ATAL-98).* Springer-Verlag, Heidelberg. pp. 1-10

Giddens, A. (1984), *The Constitution of Society: Outline of the Theory of Structuration*, Cambridge University Press, Cambridge, MA

Gilbert, N. and Troitzsch, K. (1999), *Simulation for the Social Scientist*, Open University Press, London

Gödel, K. (1965), Concerning formally undecidable sentences of principia mathematica and related systems, in: Davis, M. (ed) *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions*, Raven Press, Hewlett NY

Goffman, E. (1959), *The Presentation of Self in Everyday Life*, Doubleday, Garden City, NY

Guerin, F., Kamyab, K. and Goulev, P. (2001), *Deliverable D6.3: Software Platform Specification – Progress and Future Plans , SoNG*, CD-ROM

Guye-Vuillème, A., Capin, T.K., Pandzic, I.S., Magnenat-Thalmann, N. and Thalmann, D. (1998) 'Non-verbal communication interface for collaborative virtual environments', In: Snowdon, D. and Churchill, E. (eds) *Proceedings of the Second International Conference on Collaborative Virtual Environments (CVE'98).* ACM Press, New York. pp. 105-112

Guye-Vuillème, A., Capin, T.K., Pandzic, I.S., Magnenat-Thalmann, N. and Thalmann, D. (1999), "Non-verbal communication interface for collaborative virtual environments", *The Virtual Reality Journal*, vol. 4, pp. 49-59

Guye-Vuillème, A. and Thalmann, D. (2000) 'Requirements for an architecture for believable social agents', In: *Proceedings of the Fourth International Conference on Autonomous Agents (Agents 2000)*. ACM Press, New York. pp. 48-49

Guye-Vuillème, A. and Thalmann, D. (2001), "A high level architecture for believable social agents", *The Virtual Reality Journal*, vol.5, pp. 95-106

Guye-Vuillème, A. and Thalmann, D. (2002) 'Specifying MPEG-4 body behaviours', In: *Proceedings of Computer Animation 2002 (CA 2002), 19-21 June 2002, Geneva, Switzerland*. IEEE Computer Society Press, Washington. pp. 126-132

Hall, E.T. (1966), *The Hidden Dimension*, 1st ed, Doubleday, Garden City, NY

Hallinan, M.T. (1981), Recent advances in sociometry, in: Asher, S.R. and Gottman, J.M. (eds) *The Development of Children's Friendships*, Cambridge University Press, New York

H-Anim Standard homepage, (n.d.). <http://www.hanim.org/> (accessed 1st September 2003)

Hare, P., Borgatta, E.F. and Bales, R.F. (eds) (1966), *Small Groups: Studies in Social Interaction*, Alfred A. Knopf, New York

Hayduk, L.A. (1983), "Personal space: Where we now stand", *Psychological Bulletin*, vol. 94, pp. 293-335

Hejl, P. (1984), Towards a theory of social systems: self-organization and self-maintenance, self-reference and syn-reference, in: Ulrich, H. and Probst, G.B. (eds) *Self-Organization and Management of Social Systems*, Springer, Berlin u.a.

Hollander, E.P. and Offermann, L.R. (1990), "Power and leadership in organizations: Relationships in transition", *American Psychologist*, vol. 45, pp. 179-189

Ingrand, F.F, Georgeff, M.P and Rao, A.S. (1992), "An architecture for real-time reasoning and system control", *IEEE Expert/Intelligent Systems*, vol. 7, no. 6, pp. 34–44

ISO/IEC JTC 1/SC 29/WG11 N2502, Information Technology – Generic Coding of Audio-Visual Objects, Part 2: Visual, October 1998

ISO/IEC JTC 1/SC 29/WG11 N2739 subpart 2, MPEG-4 Version 2- BIFS, March 1999

JADE homepage, (n.d.). <http://sharon.cselt.it/projects/jade/> (accessed 1st September 2003)

Jennings, N. and Kalenka, S. (1998) 'Socially responsible decision making by autonomous agents', In: *Proceedings of the Fifth International Colloquium on Cognitive Science*, San Sebastian, Spain

Jennings, N., Sycara, K. and Wooldrigde, M. (1998), "A roadmap of agent research and development", *Autonomous Agents and Multi-Agent Systems*, vol. 1, no. 1, pp. 275-306

JESS homepage, September 4, 2003. <http://herzberg.ca.sandia.gov/jess/> (accessed 4st September 2003)

Kalenka, S. (2001), *Modelling Social Interaction Attitudes in Multi-Agent Systems*, Ph.D Thesis, Department ofElectronics and Computer Science, Southampton University

Kallmann, M., Monzani, J.S., Caicedo, A. and Thalmann, D. (2000) 'ACE: A platform for the real time simulation of virtual human agents', In: Magnenat-Thalmann, N., Thalmann, D., Arnaldi, B. (eds) *Computer Animation and Simulation 2000: Proceedings of the Eurographics Workshop (EGCAS'2000)*. Springer, Wien. pp. 73-86

Kamyab, K. (2001), *Deliverable D 6.2: FIPA Agent Platform Integration, SoNG,* CD-ROM

Kamyab, K., Guye-Vuillème, A., Kshirsagar, S., Guerin, F. and Arafa, Y. (2001), *Deliverable D6.1: Software Platform Specification – Initial Design, SoNG*, CD-ROM

Kendon, A. (1981), *Nonverbal Communication, Interaction, and Gesture*, Mouton, The Hague

Ketrow, S.M. (1999), Nonverbal aspects of group communication, in: Frey, L.R., Gouran, D.S. and Poole, M.S. (eds) *The Handbook of Group Communication Theory and Research*, SAGE Publications, Thousand Oaks [etc.]

King, W. and Ohya, J. (1996) 'The representation of agents: anthropomorphism, agency and intelligence', In: *Proceedings of the 1996 Conference on Human Factors in Computing Systems - CHI'96.* ACM Press, New York. pp. 289-290

Kirchgässner, G. (1991), *Homo Oeconomicus: Das ökonomische Modell individuellen Verhaltens und seine Anwendung in den Wirtschafts- und Sozialwissenschaften*, J.C.B. Mohr (Paul Siebeck), Tübingen

Knapp, M.L. and Vangelisti, A.L. (1996), *Interpersonal Communication and Human Relationships*, Allyn & Bacon, Needham Heights, MA

Koda, T. and Maes. P. (1996) 'Agents with faces: the effects of personification of agents', In: *Proceedings of Human-Computer Interaction'96*. Springer, London. pp. 98-103

Kshirsagar, S., Molet, T. and Magnenat-Thalmann, N. 'Principal components of expressive speech animation', In: Ip, H.S.H. et al. (eds) *Proceedings of the Computer Graphics International 2001 (CGI 2001)*. IEEE Computer Society Press, Washington. pp 38-44

Kshirsagar, S., Guye-Vuilleme, A., Kamyab, K., Magnenat-Thalmann, N., Thalmann, D. and Mamdani, E. (2002) 'Avatar Markup Language', In: Mulder, J.D. and van Liere, R. (eds) *Virtual Environments 2000: Proceedings of the Eurographics Workshop on Virtual Environments (EGVE 2002)*. Springer, Wien. pp. 169-177

Kuhn, T. (1962), *The Structure of Scientific Revolution*, University of Chicago Press, Chicago

Laird, J.E., Newell, A. and Rosenbloom, P.S. (1987), "Soar: An architecture for general intelligence, *Artificial Intelligence*, vol. 33, pp. 1-64

Latané, B. (1981), "The psychology of social impact", *American Psychologist*, vol. 36, pp. 343-356

Lester, J., Converse, S., Kahler, S., Barlow, T., Stone, B. and Bhogal, R. (1997a) 'The persona effect: affective impact of animated pedagogical agents', In: *Proceedings of the 1997 Conference on Human Factors in Computing Systems - CHI'97.* ACM Press, New York. pp. 359-366

Lewin, K. (1948), *Resolving Social Conflicts: Selected Papers on Group Dynamics*, Harper, New York

Ley, R.D. and Johnson, L.E. (1990), "Selecting social goals: alternative concepts of rationality", *American Journal of Economics and Sociology*, vol. 49, no. 4, pp. 469-482

Linton, M. (1975), Memory for real-world events, in: Norman, D.A. and Rumelhart, D.E. (eds) *Explorations in Cognition*, Freeman, San Francisco

Linton, R. (1968), *De l'Homme*, Editions de Minuit, Paris

Lorenz, K. (1950), "The comparative method in studying innate behaviour patterns", *Symposia of the Society for Experimental Biology*, vol. 4, pp. 221-268

Luhmann, N. (1995), *Social Systems*, Stanford University Press, Stanford CA

Lustick, I.S. (2000) "Agent-based modelling of collective identity: Testing constructivist theory", *Journal of Artificial Societies and Social Simulation*, vol. 3, no. 1. <http://jasss.soc.surrey.ac.uk/JASSS.html> (accessed 1st September 2003)

Maes, P. (1989), "How to do the right thing", *Connection Science*, vol. 1, no. 3, pp. 291-323

Maes, P. and Schneiderman, B. (1997), "Direct manipulation vs. interface agents", *Interactions*, vol. IV, no. 6, pp. 42-61

Mataric, M.J. (1997), "Learning social behavior", *Robotics and Autonomous Systems*, vol. 20, pp. 191-204

Maturana, H. (1980), Man and society, in: Benseler, F., Hejl, P.M. and Kock, W.F. (eds) *Autopoiesis, Communication and Society*, Campus Verlag, Frankfurt

Maturana, H.R. and Varela, F.J. (1987), *The Tree of Knowledge: The Biological Roots of Human Understanding*, Shambhala Press, Boston

McPhail, C., Powers, W.T. and Tucker, C.W. (1992), "Simulating individual and collective actions in temporary gatherings", *Social Science Computer Review*, vol. 10, no. 1, pp. 1-28

Mehrabian, A. and Diamond, S.G. (1971), "Seating arrangements and conversation", *Sociometry*, vol. 34, pp. 281-289

Mehrabian, A. (1972), *Nonverbal Communication*, Aldine Publishing Company, New-York

Mehrabian, A. (1976), *Public Places and Private Spaces: The Psychology of Work, Play, and Living Environments*, Basic Books Inc. Publishers, New-York

Merton, R.K. (1959), *Social Theory and Social Structure*, Free Press, New York

Microsoft Speech Technologies homepage, September 10, 2003. <http://www.microsoft.com /speech/> (accessed 1st September 2003)

Miller, G.R. (1976), *Explorations in Interpersonal Communication*, Sage, London

Mohamed, A.M. and Huhns, M.N. (2001), Multiagent benevolence as a societal norm, in: Conte, R. and Dellarocas, C. (eds) *Social Order in Multiagent Systems*, Kluwer Academic Publishers, Boston, MA

Monzani, J.S. (2002), *An Architecture for the Behavioural Animation of Virtual Humans*, thèse de doctorat, EPFL, Lausanne

Moon, Y. and Nass, C. (1996) 'Adaptive agents and personality change: complementarity versus similarity as forms of adaptation', In: *Proceedings of the 1996 Conference on Human Factors in Computing Systems - CHI'96.* ACM Press, New York. pp. 287-288

Moreno, J.L. (ed) (1960), *The Sociometry Reader*, The Free Press, Glencoe, IL

Morris, D. et al. (1979), *Gestures*, J. Cape, London

Moss, S., Gaylard, H., Wallis, S. and Edmonds, B. (1998), "SDML: A multi-agent language for organizational modelling", *Computational and Mathematical Organization Theory*, vol. 4, no. 1, pp. 43-70

Motion Factory homepage, 2000. <http://www.motion-factory.com/> (accessed 29th May 2001)

Musse, S.R. and Thalmann, D. (1998) 'A model of human crowd behavior: group inter-relationship and collision detection analysis', In: *SIGGRAPH 98 Course notes 80*

MySQL homepage, 2003. <http://www.mysql.com/> (accessed 1st September 2003)

Nass, C., Steuer, J. and Tauber, E. (1994) 'Computers are social actors', In: *Proceedings of CHI'94.* ACM Press, New York. pp. 72-78

Newell, A. and Simon, H. (1972), *Human Problem Solving*, Prentice-Hall, Englewood Cliffs, NJ

Noble, J. and Franks, D.W. (2002) 'Social learning mechanisms compared in a simple environment', In: Standish, R.K., Bedau, M.A. and Abbass, H.A. (eds) *Artificial Life VIII: Proceedings of the Eighth International Conference on Artificial Life*. MIT Press, Cambridge, MA. pp. 379–385

Norman, D.A. (1997), How might people interact with agents?, in: Bradshaw, J. (ed) *Software Agents*, MIT Press, Menlo Park, Calif.

Octaga homepage, 2001. <http://www.octaga.com/> (accessed 1st September 2003)

Panzarasa, P., Norman, T. and Jennings, M. (1999) 'Modeling sociality in the bdi framework', In: Lui, J. and Zhong, N. (eds) *Proceedings of the First Asia-Pacific International Conference on Intelligent Agent Technology (IAT'99)*. World Scientific Publishing, Singapore. pp. 202-206

Parsons, T. (1951a), *The social system*, Free Press, New York

Parsons, T. and Shils, E.A. (eds) (1951b), *Toward a general theory of action*, Harper and Row, New York

Patterson, M. (1983), *Nonverbal Behavior: A Functional Perspective*, Springer, New York, NY

Pelachaud, C. and Poggi, I. (2002), "Multimodal embodied agents", *The Knowledge Engineering Review*, vol. 17, no. 2, pp. 181-196

Perlin, K. and Goldberg, A. (1996) 'Improv: a system for scripting interactive actors in virtual worlds', In: *Proceedings of SIGGRAPH'96*. ACM Press, New York. pp. 205-216

Ponder, M., Papagiannakis, G., Molet, T., Magnenat-Thalmann, N. and Thalmann, D. (2003) 'VHD++ development framework: towards extendible, component based VR/AR simulation engine featuring advanced virtual character technologies', In: *2003 Computer Graphics International (CGI 2003)*. IEEE Computer Society Press, Washington. pp. 96-104

Preda, M. and Preteux, F. (2002) 'Critic review on MPEG-4 face and body animation', In: *Proceedings of the IEEE International Conference on Image Processing (ICIP'2002)*, Rochester, *NY*, 22-25 September 2002. pp. 505- 508

Prendinger, H. and Ishizuka, M. (2001) 'Social role awareness in animated agents', In: Müller, J.P. et al. (eds) *Proceedings of the Fifth International Conference on Autonomous Agents*. ACM Press, Montreal. pp. 270-277

Prendinger, H. and Ishizuka, M. (2002) 'Evolving social relationships with animate characters'. In: *Proceedings of AISB'02 Symposium on Animating Expressive Characters for Social Interactions*. AISB Publishers, London. pp. 73–78

Reeves, B. and Nass, C. (1996), *The Media Equation*, Cambridge University Press, Cambridge, MA

Reynolds, C. (1987), "Flocks, herds and schools: A distributed behavioral model", *Computer Graphics (SIGGRAPH '87 Conference Proceedings)*, vol. 21, no. 4, pp. 25-34

Rivano-Fischer, M. (1988), *Nonverbal behavior in interpersonal situations: notes on spatial behavior in public places*, Akmquist & Wiskell international, Stockholm, Sweden

Rizzo, P., Veloso, M., Miceli, M. and Cesta, A. (1999), "Goal-based personalities and social behaviors in believable agents", *Applied Artificial Intelligence*, vol. 13, no. 3, pp. 239–273

Ronald, E.M.A., Sipper, M. and Capcarrère, M.S. (1999) 'Testing for emergence in artificial life', In: Floreano D. et al. (eds) *Advances in Artificial Life: Proceedings of 5th European Conference ECAL'99*. Springer, Heidelberg. pp. 13-17

Rosenberg, B.G. and Langer, J. (1965), "A study of postural-gestural communication", *Journal of Personality and Social Psychology*, vol. 2, no. 4, pp. 593-597

Rousseau, D. and Hayes-Roth, B. (1998) 'A social-psychological model for synthetic actors', In: Sycara, K.P. and Wooldridge, M. (eds) *Proceedings of the Second International Conference on Autonomous Agents*, St. Paul, Minnepolis, USA, May 9-13, 1998. ACM Press, New York. pp. 165-172

Rubin, Z. (1973), *Liking and Loving: An Invitation to Social Psychology*, Holt, Rinehart, and Winston, New York

SAX homepage, (n.d.). <http://www.saxproject.org/> (accessed 1st September 2003)

Schwartz, B., Tesser, A. and Powell, E. (1982), "Dominance cues in nonverbal behavior", *Social Psychology Quarterly*, vol.45, no. 2, pp. 114–120

Searle, J.R. (1980), "Minds, brains, and programs", *Behavioral and Brain Sciences*, vol. 3, pp. 417-457

Sengers P. (1997) 'Socially intelligent agent-building', In: *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-97).* AAAI Press, Menlo Park, CA

Sichman, J.S. (1995), *Du Raisonnement Social chez les Agents*, thèse de doctorat, Institut National Polytechnique de Grenoble, Grenoble

Simon, F., Steinbrückner, F. and Lewerentz, C. (2000), *3D-Spring Embedder for Complete Graphs,* Technical Report 11/00, Computer Science Reports, Brandenburg University of Technology at Cottbus 2000

SoNG homepage, (n.d.). <http://www.octaga.com/SoNG-Web/> (accessed 1st September 2003)

Steiner, I.D. (1972), *Group Process and Productivity*, Academic Press, New York

Stocker, R., Green, D.G. and Newth, D. (2001), "Consensus and cohesion in simulated social networks", *Journal of Artificial Societies and Social Simulation*, vol. 4, no. 4. <http://jasss.soc.surrey.ac.uk/JASSS.html> (accessed 1st September 2003)

SWARM homepage, (n.d.). <http://www.swarm.org> (accessed 1st September 2003)

The Sims homepage, (n.d.). <http://thesims.ea.com/> (accessed 1st September 2003)

Thomas, W.I. (1923), *The unadjusted girl*, Little, Brown, and Co., Boston

Tolani, D., Goswami, A. and Badler, N. (2000), "Real-time inverse kinematics techniques for anthropomorphic limbs", *Graphical Models*, vol. 62, no. 5, pp. 353-388

Tuckman, B.W. (1965), "Developmental sequences in small groups", *Psychological Bulletin*, vol. 63, pp. 384-399

Tyrrell, T. (1993) 'Defining the action selection problem'. In: *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*. Lawrence Erlbaum, Hillsdale, NJ. pp. 1152-1157

Ulicny, B. and Thalmann, D. (2001) 'Crowd simulation for interactive virtual environments and VRtraining systems', In: *Computer Animation and Simulation 2001: Proceedings of the Eurographics Workshop*. Springer , Wien. pp. 163-170

Van Mulken, S., Andre, E. and Mueller, J. (1998) 'The persona effect: how substantial is it ?', In: *Proceedings of HCI '98*. Springer, Berlin. pp. 53-66

VHML homepage, (n.d.). <http://www.vhml.org/> (accessed 1st September 2003)

Vilhjálmsson, H.H. and Cassell, J. (1998) 'BodyChat: autonomous communicative behaviours in avatars', In: Sycara, K.P. and Wooldridge, M. (eds) *Proceedings of the Second International Conference on Autonomous Agents*, St. Paul, Minnepolis, USA, May 9-13, 1998. ACM Press, New York. pp. 269-276

Villamil, M.B., Musse, S.R. and de Oliveira, L.P.L. (2003) 'A model for generating and animating groups of virtual agents'. In: Rist, T. et al. (eds) *Proceedings of the 4th International Working Conference on Intelligent Virtual Agents - IVA 2003.* Springer, Berlin. pp. 164-169

VRML homepage, April 17, 1995. <http://www.w3.org/MarkUp/VRML/> (accessed 1st September 2003)

Wagenaar, W.A. (1988), People and places in my memory: A study on cue specificity and retrieval from autobiographical memor, in: Gruneberg, M.M., Morris, P.E. and Sykes, R.N. (eds) *Practical Aspects of Memory: Current Research and Issues*, Wiley, New York

Watkins, J. (1957), Historical explanation in the social sciences, in: Gardiner, P. (ed) *Theories of History*, The Free Press, Glencoe

Wcislo, W.T. (2000), Environmental hierarchy, behavioral contexts, and social evolution in insects, in: Martins, R.P., Lewinsohn, T.M. and Barbeitos, M.S. (eds) *Ecologia e Comportamento de Insetos*, Oecologia Brasiliensis supplement, vol. 8, Rio de Janeiro, Brasil

Weber, M. (1968), *Economy and Society*, Bedminister Press, New York

Weitz, S. (ed) (1974), *Nonverbal Communication: Readings with Commentary*, Oxford Univ. Press, New York [etc.]

Wooldridge, M. and Jennings, N. (1995), "Intelligent agents: Theory and practice", *Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152

XML Schema homepage, (n.d.). <http://www.w3.org/XML/Schema> (accessed 1st September 2003)

Appendix MPEG-4 BIFS Nodes for Face and Body Animation

Face node

The *Face node* is used to define and animate a face in the scene. In order to animate the face with a facial animation stream, it is necessary to link the *Face node* to a BIFS-Anim stream, which deals with the local or remote changes of the scene over time. The node has five fields: *fap*, *fdp*, *fit*, *ttsSource* and *renderedFace*.

fap

The *fap* field shall contain a *FAP* node, describing the facial animation parameters (FAPs). Each *Face* node shall contain a non-null *fap* field. There are 66 low level FAPs indicating the movements of the feature points of the face, and two high level FAPs, visemes and expressions. There are 14 visemes and 6 expressions defined.

fdp

The *fdp* field, which defines the particular look of a face by means of downloading the position of face definition points or an entire model, is optional. If the *fdp* field is not specified, the default face model of the terminal shall be used. The *FDP* node contains complete information of the face geometry:

- coordinate locations of the feature points
- coordinates of the texture associated with the feature points
- Face Definition Table Node: this is used to define the vertex by vertex displacement of the face object, in order to define highly individualized facial expressions. This is optional along with the following *faceSceneGraph* field, but if provided, can be used for complete animation of the face, and the decoder does not need any face deformation tool to interpret FAPs.
- The *faceSceneGraph* field contains the complete face geometry with an *IndexedFaceSet* and a *Transform* node.

fit

The Face Interpolation Table (*fit*) field, when specified, allows a set of FAPs to be defined in terms of another set of FAPs. When this field is non-null, the terminal shall use *fit* to compute the maximal set of FAPs before using the FAPs to compute a 3D mesh. The mesh is not the face model itself, but is used to define the shape of the face model.

ttsSource

The *ttsSource* field shall only be non-null if the facial animation is driven by a Text-To-Speech (TTS) source. In this case the *ttsSource* field shall contain an *AudioSource* node and the face shall be animated using the phonemes and bookmarks received from the TTS.

renderedFace

Output values describing the current facial geometry are written in the *renderedFace* field.

Body node

The *Body* node is the top-level node for a MPEG-4 body. This node has three fields, a MF3Dnode field *defaultBody*, a *bap* field containing body animation parameters (BAP), and an optional BDP field called *bdp* for body definition parameters. When the *bdp* field is empty (null), a default body model is loaded and stored in the *defaultBody* field. The *bap* field must already be present.

bap

The *bap* field is used to store the current state of the body animation. It consists of 296 integer *BAP* nodes, each one representing a certain degree of freedom value of a given body articulation standardized in the H-Anim standard (*H-Anim Standard homepage*). A *BAP* node is filled by a FBA decoder processing a FBA Stream.

bdp

The *bdp* child of the Body node contains the representation of a particular body that has to be loaded. A Body definition uses the H-Anim PROTOs: *Humanoid*, *Joint* and *Segment*. *BDP* nodes, used in the *bdp* field, can contain a non-null *bodyDefTable* field. In such a case, this field defines the way the body is deformed during an animation.

bodyDefTable

The *bodyDefTable* field defines displacements to apply to each vertex of each segment subject to a deformation in a certain BAP combination. This is a basic deformation system that lets define rough deformations as well as very precise deformations, depending on the amount of information given in this field. When the body is in a certain posture (BAP combination), all the vertices subject to deformations are displaced with an interpolated translation. The final interpolation is dependent on the displacements given for certain BAPs combinations.

PROTOs

In addition to the *Face* and *Body* nodes, PROTOs, the data structures resulting from VRML 2 prototyping mechanism, are very important for body animation in MPEG-4. For example, the representation of a Body is defined as a hierarchy of PROTO *Joint* and PROTO *Segment*. These two types of PROTOs were introduced with the H-Anim standard. For body animation in particular, joints names are vital in order to associate animation data to the 3D scene; this information is available in the PROTO *Joint*. Therefore, the MPEG-4 encoder, decoder and player shall support PROTOs. The following H-Anim v1.1 PROTOs are referenced in the *bdp* field (*bodySceneGraph* subhierarchy).

Humanoid PROTO

The *Humanoid* PROTO contains some general information, like the name of the humanoid, its sex, age, height, and weight, encapsulates a global referential that allows the positioning of the humanoid in space as well as a list of its joint and segments.

Joint PROTO

The *Joint* PROTO represents a body articulation. It contains the name of the joint but also other information, like the joint's stiffness, which is useful to specific applications, e.g. inverse kinematics. As previously mentioned, the name of the joint is very important in order to allow a body animation library to update the correct geometry. Humanoid joints' names have been standardized in H-Anim v1.1 (*H-Anim Standard homepage*).

Segment PROTO

The *Segment* PROTO encapsulates as its child a field defining the geometry of body parts. Information like *mass* and *centerOfMass* is useful for animation based on inverse kinematics. The names of the segments have also been standardized.

Site PROTO

The *Site* PROTO can be used to define sites (3D positions) where to attach various accessories to the body, like jewellery. For example, a ring geometry can be attached to a given finger using a *Site* proto.

Curriculum Vitae

Anthony Guye-Vuillème

Research assistant
Virtual Reality Lab
Swiss Federal Institute of Technology, EPFL-VRlab
Lausanne CH-1015 Switzerland
Phone: +41 21 693-5248 (lab) Fax: +41 21 693-5328
Email: Anthony.Guye-Vuilleme@epfl.ch, aguye@urbanet.ch
Web: http://vrlab.epfl.ch/~aguye/

Born May 29, 1971, in Monthey, Switzerland. Origin: Les Bayards (Neuchâtel), Switzerland. Parents: Rose-Marie and Jean-Frédéric Guye-Vuillème, Dr. Ing. chem. ETH. First language: french. Other languages: english, italian, german.

Anthony Guye-Vuillème obtained a "Licence ès sciences politiques" (BScPol) from the University of Lausanne in 1993, a "Maîtrise ès sciences sociales" (MSocSc) from the University of Lausanne in 1998, and a "Certificat de Maîtrise de spécialisation en informatique" from the Swiss Federal Institute of Technology in 2000.

Before his position at the Virtual Reality Lab, he was a research assistant at the Swiss Graduate School of Public Administration (IDHEAP). In addition to his research activities, he has been working since 1996 as a consultant for QUOD in the areas of large scale databases and intranet development.

Selected Publications

Guye-Vuillème, A., Capin, T.K., Pandzic, I.S., Magnenat-Thalmann, N. and Thalmann, D. (1998) 'Non-verbal communication interface for collaborative virtual environments', In: Snowdon, D. and Churchill, E. (eds) *Proceedings of the Second International Conference on Collaborative Virtual Environments (CVE'98)*. ACM Press, New York. pp. 105-112

Guye-Vuillème, A., Capin, T.K., Pandzic, I.S., Magnenat-Thalmann, N. and Thalmann D. (1999), "Non-verbal communication interface for collaborative virtual environments", *The Virtual Reality Journal*, vol. 4, pp. 49-59

Guye-Vuillème, A. and Thalmann, D. (2000) 'Requirements for an architecture for believable social agents', In: *Proceedings of the Fourth International Conference on Autonomous Agents (Agents 2000)*. ACM Press, New York. pp. 48-49

Guye-Vuillème, A. and Thalmann, D. (2001), "A high level architecture for believable social agents", *The Virtual Reality Journal*, vol.5, pp. 95-106

Guye-Vuillème, A. and Thalmann, D. (2002) 'Specifying MPEG-4 body behaviours', In: *Proceedings of Computer Animation 2002 (CA 2002), 19-21 June 2002, Geneva, Switzerland*. IEEE Computer Society Press, Washington. pp. 126-132