D 4.1

STATE OF THE ART REPORT AND REQUIREMENT SPECIFICATIONS

Project Number	FP7-ICT-231824
Project Title	Integrating Research in Interactive Storytelling (NoE)
Deliverable Number	D4.1
Title of Deliverable	State of the Art Report and Requirement Specifications
Workpackage No. and Title	WP4 - Hybrid Intelligent Virtual Actors
Workpackage Leader	INRIA
Deliverable Nature	Report
Dissemination Level	Public
Status	Final
Contractual Delivery Date	30th September 2009
Actual Delivery Date	30th September 2009
Author(s) / Contributor(s)	Stéphane Donikian, Nicolas Pépin (INRIA) Paolo Petta (OFAI)
Number of Pages	65

Table of Contents

Abstract

Populating a virtual world with autonomous agents is a complicated task with many different approaches and problems which are given a thorough overview in this report. These problems can range anywhere from complex abstract reasoning to subtle animation of the fingers during a grasping motion. Modelling believable autonomous agents also needs to take into account many different aspects from very different disciplines, ranging from cognitive psychology to mechanics. This deliverable presents a state of the art on virtual humans, especially on their behaviour description and simulation. This report covers the work done in both, cognitive science and computer graphics.

1. Introduction

Virtual Humans have at least two usages in a virtual reality environment, the first one is to represent a real human in the virtual world and it is then called an avatar, and the second one is to live autonomously inside the virtual environment, that is to say, perceive, decide and act on its own. Both uses require a geometrical representation of the human, but in the second case it is also necessary to model the behavioural activity instead of reproducing the activity of a real character.

Behavioural models have been introduced to design autonomous entities such as living beings or robots. In this report, we focus on anthropomorphic characters, and we will compare different formalisms introduced in computer science over the last twenty years to theories proposed by cognitive scientists. In computer science, the goal in modelling human behaviours is not to reproduce the complexities of the human brain and body but to propose software architectures able to reproduce believable human behaviours in dedicated activity contexts. To model a human behaviour, it is necessary to address different issues, such as multi-sensory perception, memory activity, face and body muscular control including emotion expressivity, and action selection. In short, it is necessary to investigate the operation of various faculties that constitute together the human mind, without forgetting their relation with the body.

In complement of the study of these general mechanisms underlying any human behaviour, the work should also be concerned with the study of human faculties in dedicated activities, such as navigating in a city, using a work instrument or conducting a structured interview. The comprehension of human behaviours requires competences in fields as varied as the neurosciences, psychology and behavioural biology. There are two approaches regarding the study of human behaviour: the first one is system oriented and consists in using the work carried out in the neural sciences to look at the brain activity of patients subjected to various stimuli, according to well defined operating protocols. The techniques used are for example brain imaging: PET (Positron-Emission Tomography), fMRI (functional Magnetic Neuro Imaging) or the measurement of electric activity: ERP (Event Related Potentials). It focuses mainly on concepts of signal transmission in networks, control and state feedback. The second, more symbolic, approach consists in modelling the human behaviour in a more abstract way by the way of modules, each of which describes a mechanism. These modules are possibly hierarchically structured and are linked by sequencing and/or parallel relations.

The two approaches have different advantages: the first, closer to the neuro-physiological data, is suitable for the modelling of neural and sensori-motor activities, while the second approach makes it possible to abstract from the biophysical processes within the brain and to propose a modelling of the behaviour based on competences. None of the models proposed in either of the two approaches is completely satisfactory to model the human behaviour in its entirety. However, we are still more interested in the second approach, due to its more macroscopic vision and to its similarity with software component architectures. Indeed, our problem is not to reproduce human intelligence but to propose a software architecture allowing to model credible behaviour of virtual anthropomorphic actors evolving in real-time in virtual worlds. These latter ones can represent particular situations studied by psychologists or even correspond to an imaginary universe described by a scenario writer. In any case, the proposed architecture should mimic relevant human intellectual and physical functions.

An important bibliographical study made in recent years [Donikian 2004] exposed the diversity, even the antagonism, of the approaches in the field of cognitive science and the substantial lack of federation models to propose an architecture allowing to connect together the various functions used in the human behaviour even for the simplest ones. The various approaches focus generally on one or the other of the functions or on a particular method of their relation.

Von Uexküll [Uexküll56] defined the environment as the part of the outside world with which a human or animal body can naturally interact. The human body is in constant interaction with its environment by means of sensors and of effectors, as illustrated in Figure 1, below.

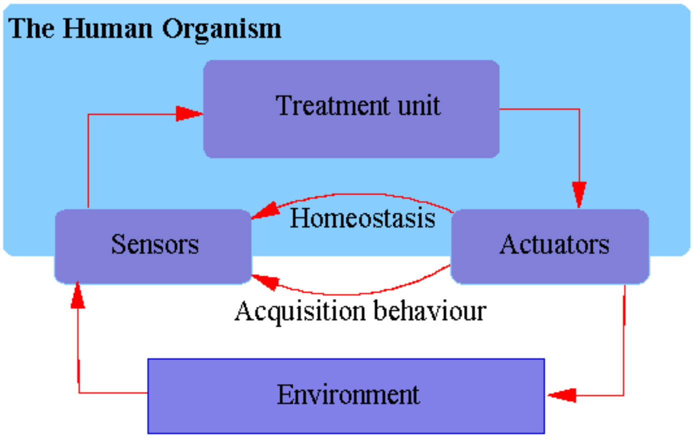

Figure 1: The human organism and its environment.

The overview of the system is a perception-decision-action loop. The first arrow from actuators to sensors is called *homeostasis* and designates the internal regulation feedback used by the body for the preservation of the biological parameters against variations of the ambient environment, whereas the second arrow stands for *acquisition behaviour,* which comprises actions driven by perception (e.g. turning the head to see something which should be on side). However, the most important loop corresponds to the interaction with the environment. H Mallot [Mallot99] classifies human behaviour into various levels of complexity and illustrates four, as shown in Figure 2.

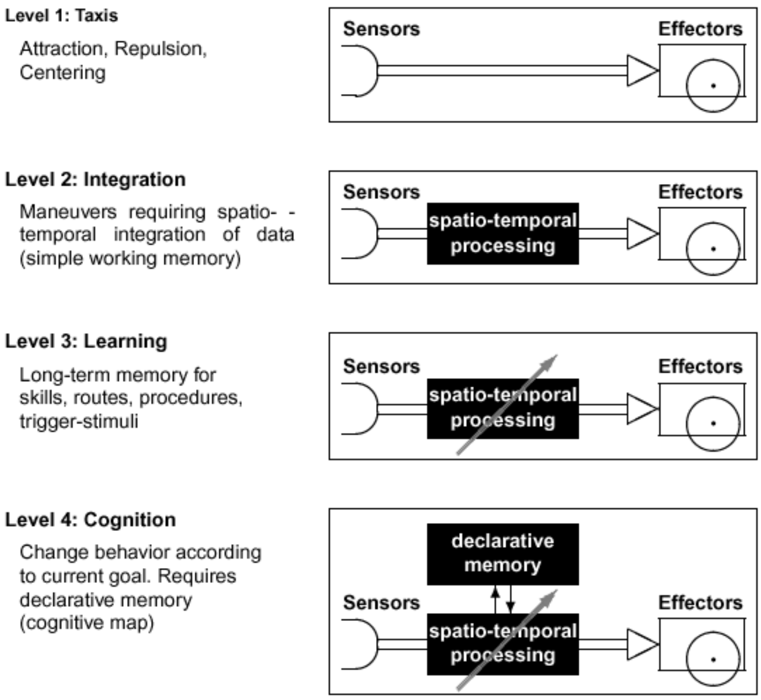

Figure 2: The four levels of behaviour introduced by Mallot in fig. 1, pp 41 of [Mallot99].

The first level represents reflex behaviour of type attraction/repulsion and can be defined simply by an interconnection between sensors and effectors. The following level represents behaviour which requires inter-neuron integration in hidden layers and which allows to describe behaviour of manoeuvre type, requiring spatio-temporal integration. The third level deals with the plasticity of spatio-temporal integration, but this learning behaviour still remains completely determined by the state of the sensory data. The fourth level, dedicated to cognitive behaviour, does not depend any more on sensory stimuli only, but also on a current *purpose* pursued by the person.

Trying to reproduce human behaviour requires developing formalisms to model and then systems to simulate autonomous anthropomorphic characters. No theory exists for determining either the necessary or sufficient structures needed to support particular capabilities, and certainly not to support general intelligence. In fact, it is a basic tenet in the cognitive sciences that the so-called identification problem, choosing the one out of several candidate models that best describes a given set of results, is unsolvable in principle [Fum07, Anderson76]. As direction and inspiration towards the development of such a theory, Newell [Newell90] posits that one way to approach sufficiency is by modelling human cognition in computational layers or bands: reactive, cognitive, rational and social. Even if researchers do not agree on the perimeter of each band, there is however a broader agreement on the decomposition into several layers going from very low level control loops, providing very fast responses to stimuli (sensory-motor control), to higher levels such as the cognitive one manipulating and reasoning on symbols, and the social one including personality, emotions and relations between humans.

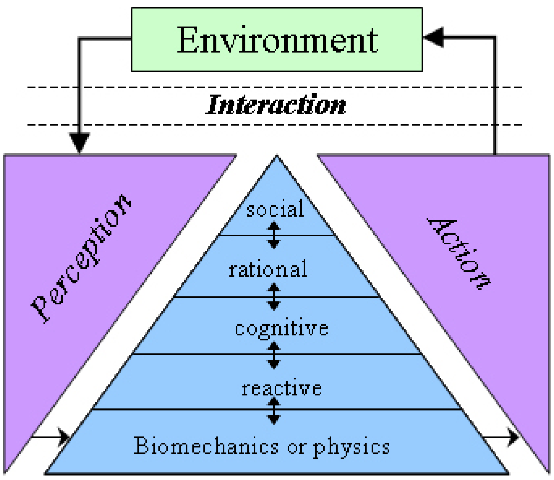

Figure 3: The Behavioural Pyramid [Donikian04].

Figure 3 refer to the classification made by Donikian in [Donikian04]. In this classification, the reactive layer is in charge of the realization of simple behaviours that do not need to explicitly manipulate an abstract representation of the world. The cognitive layer manages the abstract knowledge representation system. At this level, perception is interpreted under the form of symbols enabling reasoning. The rational layer manages reasoning on symbols provided by the cognitive layer. It includes goal oriented action selection, planning and inference capabilities. The social layer includes personality, emotions, relation between virtual humans, verbal and non-verbal communication.

In this report, we will first present both theories coming from psychological and cognitive sciences, and models proposed in computer sciences. We will also try to compare models to theories to try to characterize the quality of obtained behaviours in terms of credibility or believability.

2. Reactive Behaviour Models

2.1 First Generation

One can place historically the first works on behavioural animation at the end of the 1980s with in particular the article by C. Reynolds on the animation of flocks of birds [Reynolds87]. A first set of approaches was studied in parallel in the literature, to the middle of the 1990s, concerning the definition of the decision-making part of the behavioural model:

- **Sensor-Effector network:** This approach defines the behaviour of objects from a set of sensors and effectors interconnected by a network of intermediate nodes transforming the passing-by information [Wilhelms89, Panne93, Ngo93]. This type of approach includes also neural network models [Travers88]. The way an object behaves depends on the perception which it has of its environment and the way this perception is passed on through the network to the effectors which produce the movement of objects. This approach has the advantage of being able to generate a very important quantity of different movements (choice of the set of parameters) but stays on the other hand at a very low level of abstraction. Furthermore, this kind of model functions like a black box in which it is impossible to modify the slightest parameter without having to resume the complete process of configuration. Furthermore, the percentage of good controllers decreases in an exponential way with the growth of the number of parameters to be taken into account; it is then misleading to try defining a complex behaviour with this approach.

- **Behaviour rules:** as the previous approach, the approach by rules of behaviour takes as input data information conveying a certain perception of the environment and produced as output a control over the motion of objects. Here, the behaviour of objects is defined by a set of rules [Reynolds87]. The possible behaviour of an object can e.g. be represented by a decision tree, with every branch representing a different behaviour. An action satisfying the conditions of the current environment will be chosen by the application of a tree traversal algorithm. Behaviours allowed by this approach are of a higher level of abstraction than the previous one. The heart of the problem in this approach lies in the weighing of the various behaviours. The simplest solution consists in making an implicit choice on the order of rules, but this solution does not allow the specification of complex behaviours. Weighing the different branches of the tree allows not to privilege always the same rule, whereas the consideration of experts' hierarchies allows to confront several concurrent behaviours, the final choice staying at the upper level of the tree [Coderre88, Tu94].

- **Finite Automaton:** the automaton defines the various possible sequences of behaviour [Magnenat-Thalmann90]. This approach finds very quickly its limits with increasing complexity of the behaviour to be modelled. While modelling the behaviour of a car driver, Booth et al. [Booth93] have shown the necessity of using several state-machines, whose concurrent execution needs to be coordinated.

The report that one can make on these various efforts is that they are specific models conceived to be applied to particular cases, in which objects and their environments are relatively simple and perception and action capabilities are limited. Besides, there is a big disparity in the behaviour allowed by each of the approaches. Either the level of abstraction is very low and the only behaviours that may be specified are reflex-like ones (sensor-effector approach), or the level of abstraction is higher, but then the environment is necessarily reduced, completely defined, and it is then the perceptions and actions of the entity that are of a very low level of complexity. These models remain in any case relatively simple, with limited fields of perception and action, and furthermore do not take the temporal aspect into account, even though it is essential.

2.2 Second Generation

To match the challenge of higher levels of the decision-making complexity, it is necessary to handle collectively the continuous and discrete aspects, to coordinate concurrent behaviours and to manage their organizational structure. That is why the first two approaches were rather quickly abandoned in favour of an approach based on state-machines in their parallel and hierarchical versions:

- stacks of state-machines (EPFL, Switzerland) [Noser97];
- sets of communicating state-machines (PaT-Nets) (University of Pennsylvania, the United States) [Badler97];
- hierarchies of parallel state-machines (HCSM) (University of Iowa, the United States) [Ahmad94];
- parallel transition systems organized into a hierarchy (HPTS) (IRISA, France) [Donikian95].

This kind of approach is now also common in the animation and game industries. More recently, models have been proposed to handle uncertainty either by using Bayesian programming [Hy04] or decision networks [Yu07], which are a generalization of Bayesian networks. Even if these models are hierarchical in their structure, they do not allow managing the coordination between concurrent behaviours, as only pure parallelism without any relation between parallel decision networks can be managed.

As an example, let us look in more detail at the Hierarchical Parallel Transition System (HPTS). HPTS is a reactive behaviour description model combining the state-machine and the multi-agent approaches. Every state of every automaton is an agent possessing an internal state, being able to receive stimuli and to emit them in reaction. Any active state of the system receives a stream of input data, delivers a stream of output data, and possesses a continuous and discrete control. A state of the system is either a terminal state, or a composite state. G. Moreau [Moreau98] proposed an implementation of the HPTS model, including a programming language and a compiler allowing to generate equivalent C++ code. Each state has its own activity status with three possible values: {*active, suspended, inactive*}, and dedicated functions are executed when the status changes: {*begin, end, wait, resume*}. The control parameters allow to influence the behaviour to be adopted by either external or internal decisions.

A filtering function manages the coherence of the actions proposed by the active states of the parallel sub-state-machines. When concurrent behaviours are proposed by different sub-behaviours, this function is in charge of making a choice and to deliver a unified behaviour to the upper layer. An automaton is always executed after its child, allowing to select a behaviour and to mix the propositions supplied by its sub-components. So, concurrent behaviours can work in parallel and be arbitrated. This model was successfully used to model the behaviour of car drivers and pedestrians [Donikian99, Thomas00]. This language was then extended [Donikian01] to manage random choice between different possible behaviours with weighting rates and to manage also the reaction times between decisions and actions.

3. Competitive and Cooperative Approaches to Action Selection

3.1 Introduction

According to Clancey [Clancey02], goal directed behaviours are not always obtained by inference or compilation, but some actions may simply reproduce cultural motives while still others are coordinated without deliberation, just based on attention and adaptation. A person does not perform several tasks strictly in parallel, but several tasks are going to take place by merging attentively several parallel interests. It is useful to offer a mechanism allowing to execute different behaviours in parallel without having to program the synchronization of their execution by hand. Moreover, while describing a behaviour it should not be necessary to take into account all the possible behaviours that may interfere with it, as this may evolve depending on the context. Two kinds of action selection algorithms exist: the cooperative approach and the competitive approach.

3.2 Cooperation vs Competition

The cooperative approach allows combining several potentialities of action, whereas competition pursues only a single action out of all the potentially practicable ones. Because of the expressiveness of the combinations of the actions to be supported, the former approach is mainly based on the usage of arithmetical functions on quantitative data, whereas the latter approach, typically determines the best action to realize in a given context through the use of cost functions coupled with information of a more qualitative nature.

HPTS proposes both approaches through the notion of integration functions available at each level of the hierarchy. Integration functions are in fact programmable functions, and the programmer has the choice at each level to implement a competitive or a cooperative function, according to the nature of the behaviour to be handled.

3.3 Situation Calculus

In the competitive approach, one of the earliest formalisms employed to describe cognitive systems is situation calculus [McCarthy69], which allows to reason about the world and its changes. The situation calculus allows the exploration of all the possible worlds that can result from the execution of one or many actions, starting from a given world state. The reasoning in the situation calculus is based on four concepts:

- **Situations**: A situation describes the complete state of the world, at a given time, through facts that describe the properties of the environment. They can be used to deduce further facts that either hold in the present or would hold in the future;
- **Fluents**: Fluents describe properties of the world that can change with time. They are functions which take a situation as input and return the state of the property. Fluents can be used to describe an atomic fact or relations holding between objects in the environment;
- **Actions**: Actions allow changing a situation as long as a set of properties, or pre-conditions, can be verified to hold. An action modifies a situation to create a new situation, thus modifying the description of the world;

- **Knowledge**: Knowledge can be gathered for example from the world through actions that describe perceptions, which allows reasoning on a rather high level of abstraction. This can e.g. enable the gathering of information necessary to achieve a specific goal.

Using sound and complete inferencing mechanisms that build on these concepts, it is possible to derive all possible evolutions of a given world, and to infer correct sequences of actions suited to accomplish a specific goal. However, this approach suffers from one main problem, called the frame problem [Scherl03]: the fact that along with a specification of changes it is also necessary to explicitly describe everything in the world that does not change when an action is undertaken. Such a complete description is almost always impossible to achieve. To solve the frame problem, it is necessary to make the strong hypothesis of evolving in a completely closed world which assumes that everything not directly concerned by the action, remains unchanged. Situation Calculus remains however a powerful mechanism for environments of limited size, where the author can have a good control over the impact of each action. J. Funge has proposed an implementation for virtual environments through the Cognitive Modeling Language (CML) [Funge98], demonstrated with prey-predator examples[1].

3.4 ASM: Action Selection Mechanism

Still concerning the competitive approach, action selection algorithms were proposed in the field of multi-agent systems, most being extensions of the Action Selection Mechanism (ASM) algorithm introduced by P. Maes [Maes90]. We can quote the algorithm by V. Decugis and J. Ferber [Decugis98] that addresses the interesting problem of how to combine real-time reactive and planning abilities in the same model (in the particular case of robotics) in the tradition of the Brooksian Subsumption Architecture [Brooks85]. To this end, Decugis and Ferber proposed a hierarchical ASM in which the lowest levels concern basic reactive behaviours and the higher levels integrate behaviours of increasing complexity (in the same way as in the hierarchical parallel state-machine approach). At every level, a mechanism of arbitration must be used to choose among the actions proposed by the parallel ASMs.

B. Rhodes proposed another hierarchical extension of the ASMs called PHISH-Nets [Rhodes96]. This model supports the use of parameterised actions and allows to define relations between actions which are either of conflicting type or of antecedent type. These models allow to perform reactive planning, but a main inconvenience concerns the requirement of an exhaustive specification of all possible interactions between actions. Furthermore, in these models circumstances demonstrably do occur, for which no (always valid) decision function can be specified.

[1] See e.g. [Russell03] for an overview of more recent evolution of research on the frame problem (including the solution proposed by Ray Reiter and its decomposition in sub-problems).

3.5 ABL: A Behaviour Language

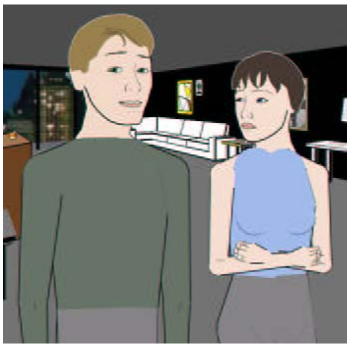

Figure 4: Image taken from Façade [Mateas02].

Façade, developed by M. Mateas and A. Stern [Mateas02] integrates in the same application the management of the structure of the dynamically evolving story, the control of the behaviour of two characters, and natural language processing for the interaction with a user interpreting the role of a visitor, as the third character of the story. Grace and Trip, the computer-controlled troubled couple in their thirties, are the protagonists of the story modelled loosely after Edward Albee's "Who's afraid of Virginia Woolf" The behaviour of the protagonists is defined by means of the reactive planner language ABL [Mateas02b]. ABL supports the specification of behaviour coordination and further includes a resource reservation mechanism allowing a behaviour to request the use of a physical resource with a certain priority.

3.6 HPTS++

F Lamarche [Lamarche02] suggested automating the mixing of behaviours with respect to their relative importance, by using the cooperative approach. To illustrate the problem addressed in this work, let us take a concrete example: a person is in front of a table, she reads a book while drinking a coffee and smoking a cigarette. This sentence describes a behaviour that is relatively simple for a human being, however in term of behavioural animation, it raises a range of problems. On one hand, it consists of three *independent* behaviours which take place simultaneously: to read a document, to drink a coffee, and to smoke a cigarette. In the previous models, describing this type of composited behaviour is relatively difficult, as it is necessary to be able to rank the component behaviours so as to meet a certain number of constraints, such as not to drink the coffee while having a cigarette in the mouth, nor to manipulate the pages of the book while the hand holds the cup of coffee. On the other hand, in the coordination of these three behaviours described separately it is necessary to avoid the re-coding of specific behaviours dedicated to their simultaneous realization.

The objective is to be able to launch the sub-behaviours and to have them be executed together automatically according to the circumstances, the social, psychological, and physical envy and physical constraints of the virtual human. The biggest synchronization problems come from the use of the internal resources of the agent: the gaze, the hands, the feet, or more generally any dependence, physical or not, limiting the parallel execution of different behaviours. To reach the objective, three new notions were introduced within the HPTS model: resources, priorities and degrees of preference, giving rise to the HPTS++ model [Lamarche02]. The mutual exclusion of the use of resources allows to define what behaviours are compatible at all times. The priority is a coefficient which indicates the importance of a behaviour in a given context. This coefficient can characterise either the adequacy (activation) or the inadequacy (inhibition) of the behaviour in a given environment. The priority can be defined to dynamically change over time.

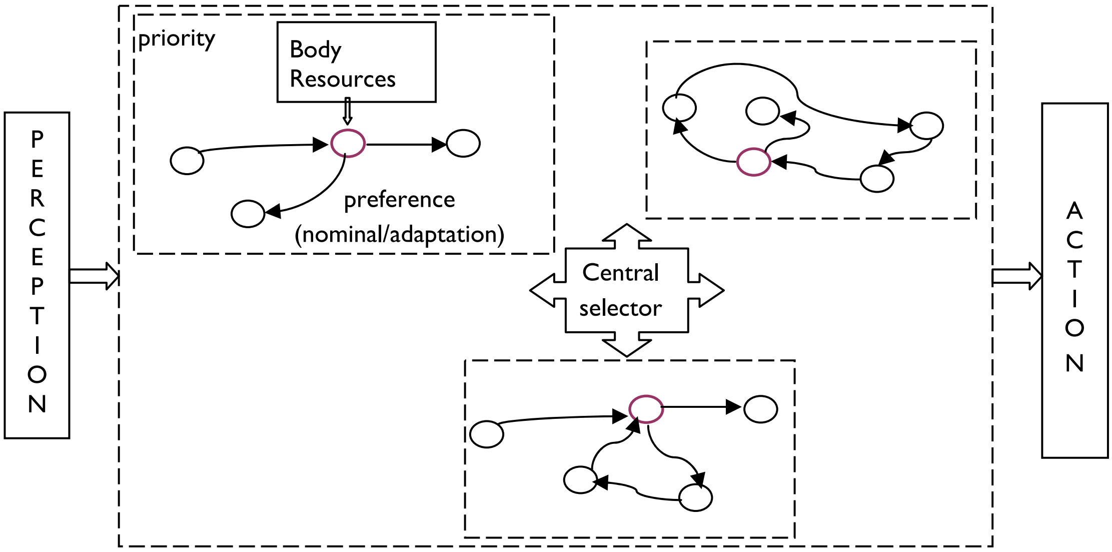

Figure 5: Coordination of concurrent reactive behaviours with HPTS++.

The degree of preference is a numerical value associated with each transition of a state-machine. This value allows to describe the tendency of the behaviour to use this transition when the associated condition is true. Thus, depending on its value, this coefficient has different meanings. If the value is positive, the transition favours the realization of the behaviour. If the value is negative, this transition does not favour the realization of the behaviour, but allows to describe a way to adapt the behaviour while releasing some resources needed by another behaviour or to coherently terminate the behaviour. A null value will not influence the behaviour.

This system allows to describe behaviours with all their adaptation possibilities at fine grained level,. , Thanks to a mechanism of interblocking avoidance, the description of a new behaviour does not require knowledge of behaviours already defined. In this way, the coordination of all active behaviours becomes generic and automatic (Figure 5). A complete description of this automatic coordination algorithm is given in [Lamarche02], with a full illustration by the example of the {*reader / drinker / smoker*} character.

4. Multi-agent Systems

4.1 Introduction

The term agent is used in a rather loose way in many domains. Even so, we can propose a minimal common definition:

> *We call agent a real or abstract entity which is capable of acting on itself and its environment, which has a partial representation of this environment, which can communicate with the other agents of a multi-agents universe, and whose behaviour is the consequence of its observations, its knowledge and its interactions with other agents.*

We distinguish two kinds of agents: reactive agents and cognitive agents. Reactive agents are not necessarily intelligent agents individually but their global behaviour can be intelligent. A frequent example of emergent stigmergic organisation is the one of the ant-hill. While all the ants are situated at equal level and none of them possesses power of authority, through the environment, actions of ants coordinate so that the colony can survive. In this way, they master complex problems such as the search for food, taking care of their eggs, the construction of nests, and reproduction.

Cognitive agents are agents that can cooperate to solve a problem. They maintain and share explicit, prospective hypothetical world models. In this case, the problems of cooperation are similar to those of small groups of individuals, who have to coordinate their activities and are sometimes brought to negotiate between them to solve their conflicts. The analogies are social and many researchers in this domain are inspired by work in sociology and in particular on the sociology of organizations and small groups.

Agents are *motivated* actors who possess purposes and knowledge of the environment. They work differently according to their type. Reactive agents possess mechanisms of reaction to the events, taking into account neither explicit description of the goals, nor explicit mechanisms of planning, but can nevertheless solve complex qualified problems. Cognitive agents have higher levels of flexibility and autonomy (i.e., “self-determination”): they possess manipulable models of the tasks to be carried out of their environment, the knowledge, expertise, and information necessary for the management of the interactions with other agents and with their environment. Such agents are known as deliberative because they possess purposes and explicit plans allowing them to pursue their goals.

4.2 BDI: Belief-Desire-Intention

Since the 1990s, agents have been a favoured metaphor for research in artificial intelligence emphasising the development of complete system designs over the focus on specific independent capabilities. Triple-tower and triple-layer designs at various resolutions and with different degrees of constraints on information flow (e.g., the class of so-called “omega-designs” as in J.P. Müller’s InterRAP [Fischer96] or pandemonium architectures such as I.A. Ferguson’s TouringMachines [Ferguson92] highlighting concurrency) have been proposed. As an example of a more complex design aimed at displaying also non-functional states typical of human behaviour, A. Sloman has been developing the H-CogAff architecture of an intelligent agent structured at several levels of abstraction, integrating several paths between perception and action, working memory, attention filters, and parallelism between tasks [Sloman97].

Of particular relevance to the present purposes, the shift in focus onto whole designs led to re-appreciation of the place and role of deliberation and planning in architectures (cf. also the already mentioned insights contributed by W. Clancey) and for renewed interest in conceptualisations of *practical* reasoning (e.g. [Pollack92]).

A seminal proposal developed by Bratman and colleagues [Bratman87, 88], builds on the key concepts of beliefs, desires and intentions, along with the issue of conditions and policies for formation and reconsideration of beliefs and intentions (see below). The beliefs capture the agent's empirical and derived knowledge about its world. They add up to an incomplete model, e.g. because of the limitations of the capacities of perception of the agent, its resource boundedness, and the impact of the subjective motivations how to go about the dynamics of the relations with the world. The desires capture persistent preferences, which influence the calculation of subjective utilities used in decisions, including decisions what courses of actions to subscribe to. The current intentions gather all the plans being pursued by the agent to satisfy and maintain satisfaction of its desires. When an agent wants to satisfy a particular desire, it searches for means that it can expect to bring about a state of affairs that qualifies as meeting corresponding desirable conditions. Herein, *plans* play a particular role as (relatively) stable commitment devices that provide particular frames of context for subsequent activities. A plan is characterised by being an incomplete (e.g., high-level, abstract, lacking details) structure that enables societies of agents to interoperate (by communicating about shared plan structures and enabling agents to model and anticipate further behaviour, thereby e.g. justifying early investments in collaboration) and individual agents to cope with the challenge of open worlds. The basic model of functioning of a rational BDI-agent is [Wooldridge00]:

Infinite loop
 Observe the surrounding world;
 Update the internal model of the world;
 Deliberate on what has to be the next desire to realize;
 Reason to find a plan to satisfy the desire;
End of loop.

Different functions are introduced, which act on the three basic concepts. The *belief revision function* modifies the state of beliefs based on new perceived data; other functions are defined for the filtering of the intentions or the planning of action. M. Wooldridge describes in [Wooldridge00] a BDI logical formalism entitled LORA (Logic of rational agent) that extends the BDI logic introduced by Rao and Georgeff [Rao92] by adding the consideration of time and the effects of the actions. It is about purely formal models which remain restricted in the abstract logical modelling of the world and which are very remote from the embodied situated cognition (cf section 6.6). The theoretical rigour offered by such logical approaches is certainly appealing and likely a necessity for safety-critical application settings. However, the clearly circumscribed target domain of BDI theory must always be taken in consideration. More generally, if the goal of computational research should be to model *human* behaviour, it is important not to equate particular logic formalisms with human reasoning (see e.g. [Staller00] for a related review).

Recently, Peinado et al. [Peinado08] proposed a narrative reformulation of the BDI theory, in order to model intelligent characters. The Narrative BDI model is a high-level narrative-oriented extension to the BDI approach. Operators used in the BDI model are under the control of the planner, thus feelings are not purely reactive but are updated by voluntary acts decided by the planning according to character's predefined personality.

4.3 PECS: Physical, Emotional, Cognitive, Social states

The PECS model [Schmidt00] was developed for the simulation of human behaviour in social contexts. This model is presented by its author as an alternative to the BDI model. The objective of PECS is to take into account variables about Physical, Emotional, Cognitive and Social states (hence the acronym), in order to cover all these behavioural levels. The architecture can be divided into three different horizontal layers (cf Figure 6).

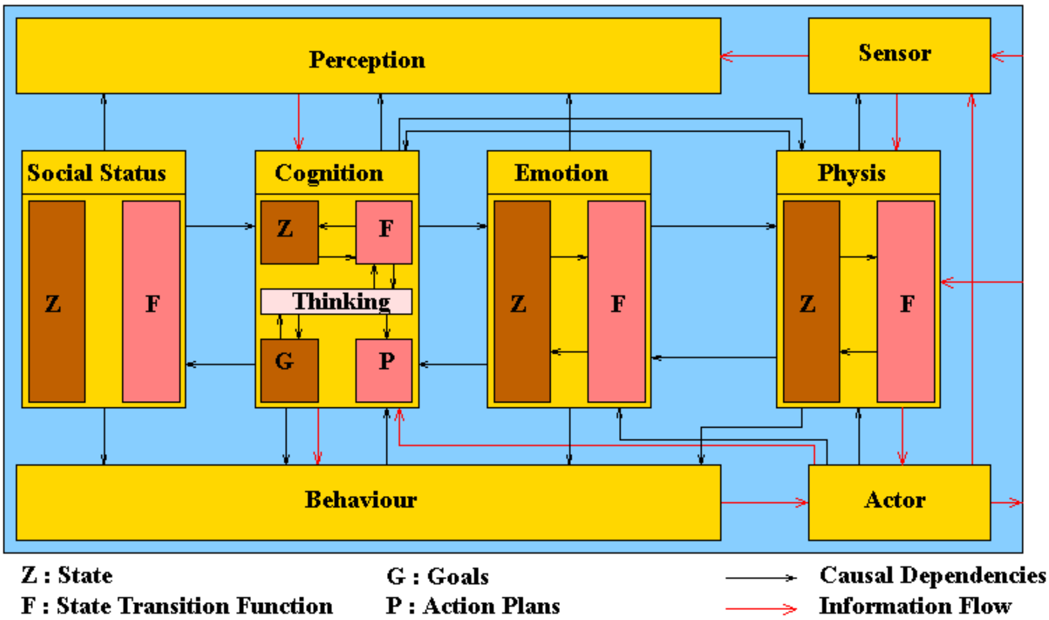

Figure 6: The layered structure of the PECS agent architecture, comprising input (top), internal (middle), and output (bottom) layers. The input and output layers are interfaced to the environment to the right of diagram (not shown in the figure).

The input layer comprises sensor and perception components and is responsible for processing of input data from the environment of the agent. The internal layer is constituted by the physics, emotion, cognition, and social status components and corresponds to the modelling of the internal state of the agent. Finally, the output layer is formed by the behaviour and actor components. The author does not go into the details of the models of each component, as the aim of the work is a domain- and theory independent conceptual design for the interaction of major operating principles within an integrated architecture of a whole agent [Urban01].

For this reason, the presented examples remain far from covering the full complexity of the announced objectives. A first case study, called Adam's model, describes the behaviour of a lonely character evolving in a universe constituted by a 12x12 two-dimensional grid, in which it has to find food sources and to avoid boxes marked as dangerous. Its actions are relatively limited: to plan, investigate, examine, walk, eat found food and to escape. A second model deals in an abstract manner with the notion of social groups, and the way agents interact to form social groups, work together, and leave a group. The modelled actions of an agent accordingly are: to choose a group, join this group, start a group, leave a group, and to disband a group.

4.4 BRAHMS

Brahms [Sierhuis01, Acquisti03] is an agent oriented language used to model and simulate human activities. This language arises from work on activity theory by W. Clancey [Clancey98, 02]. Every agent can belong to several groups. By defining the membership of an agent in a group, the group benefits from the behaviour of the agent and from those connected to the domain of activity of the group. Objects are representations of artefacts in the world that have no specific behaviour, thus they react only to state modifications of the world. Each agent has a set of beliefs that can be changed by events, such as the result of reasoning or the realization of an activity. Beliefs are linked to specific agents, whereas facts are states of the world known to all agents. The activities of agents can be decomposed into sub-activities. An agent pursuing a sub-activity is always also engaged in the meta-activity containing it. Mechanisms of suspension and resumption of an activity exist. Every activity takes time to carry out, and the duration of an activity can be fixed, chosen randomly, or parameterised by the result of reasoning over the beliefs of the agent. There are unitarian or primitive activities which possess several characteristics [Sierhuis03]. A conditional state is associated to every activity. If the conditions of a rule are part of the current belief set of an agent, then the activities which are connected to it are realised.

The working environment allows to describe such conditions. Frames of thought allow to model the progress of reasoning and to describe the consequences related to preconditions, thereby offering the means to deduct new beliefs for the agent. The operating principle is the same as for the working environment, except that no activity is associated with the conditions. The geography gives a description of the places in which the agents may carry out their activities, as well as of the possible routes between places. A change of place is made by a motion activity. A membership relation allows to describe hierarchical structures of places. An agent can also carry objects by using the *take* and *put* activities.

While cognitive modelling tools (see section 7) provide detailed models of individual cognitive tasks, Brahms focuses on how informal, circumstantial, and located behaviours of a group of individuals interact and where communication and synchronization occurs, so that the task contributions of people and machines flow together to accomplish goals. A direct link to specific virtual embodiments and situating worlds, in which the actors could evolve and develop their own activities is not provided.

4.5 Conclusion

The modelling of human behaviour in artificial intelligence is a vast field of study. However, specific models developed are difficult to apply to autonomous virtual humans, because the priority is typically put on reasoning and decision-making, to the detriment of grounded behaviour integrating realistic situated sensori-motor constituents. All the proposed models suffer from what we could call the disembodiment of the actors, as well as the very abstract vision of the world surrounding them. Indeed, the biggest weakness of this kind of models is the very frequent disregard of the environment and of the capacities of perception and action of the actor[2]. Furthermore, these models usually do not satisfy the real-time constraint of the virtual environments.

[2] As clear evidence for the importance of this issue, see the impact of the first engagement of A.Sloman in the development of a sophisticated grounded cognitive robotic architecture in the context European CoSy project in terms of forced revisions of basic principles of his architectural designs developed over the past decades.

5. Common behaviours

5.1 Introduction

In everyday life, several behaviours are used by any human character, such as walking, looking at objects, avoiding obstacles, planning a path to reach a destination. All these behaviours, apparently simple for a human being, have been the topic of substantial research efforts, described in the following sections.

5.2 Perception

In order for a virtual human to perceive its environment, it has to be given a way to sense it. Some techniques try to emulate or simulate human senses, and gather geometric information from the environment. The most studied sense is vision, since it is the sense that allows us to gather important information from the environment. Hearing has also been studied to some extent. On the other hand, the other senses (touch, smell and taste) have not been studied as extensively. Mere emulation of senses can sometimes be insufficient and more abstract and conceptual information needs to be gathered. To that effect, some techniques take advantage of the technical possibilities of providing additional interfaces to virtual environments and *cheat* by allowing the agent to perceive the environment in a way that would have been impossible if it were situated in the real world.

The most direct way to simulate vision is the top-down approach consisting in image recognition and analysis, planning and learning. These processes are aimed at extracting information from the environment through geometric analysis and comparison with knowledge stored within the agent. This approach can be very costly and, to our knowledge, has only been used in the field of robotics but not within real-time simulations of virtual humans. Even so, such an approach would eventually break down once interaction with complex machines should be required, since it is usually very hard to infer the way machines work through observation alone.

Chopra et al. developed a vision based system [Chopra99] managing visual attention and identifying two separate modes of vision: idle and deliberate. Idle perception consists in generating eye movements whenever the agent is not actively looking at something. Instead of having a fixed gaze, a walking agent looks at the horizon or at its destination and occasionally glances at the ground. When crossing a road, it will look at the traffic light more frequently. This is also known as monitoring. On the other hand, deliberate gazing focuses the agent's visual attention to the current active task. The agent's visual attention is directed towards the concerned target when reaching and grasping an object. Visual search is also present in the system, and several intermediate gazing positions are introduced between the current position being fixed by the agent and the target position. This produces the effect of the agent looking around for the target instead of looking at it directly when asked to locate it. Peripheral vision is also managed, and the agent will sometimes shift its attention to moving objects in the perimeter of its field of view. Finally, spontaneous looking directs the agent's gaze to objects that are likely to be informative or significant.

Idle vision based techniques concentrate on gaze generation and passive vision. This allows an agent wandering around an environment to look at points of interest within that environment, instead of having a fixed gaze. Courty et al. [Courty03] and Peters et al. [Peters03] use saliency maps to this end. The scene is rendered from the agent's perspective, and several filters are applied to it, allowing to isolate geometric points of interest (salient points) that stand out from the environment. The agent's gaze is then directed to these geometrically interesting points if it is not actively looking for something. This idle gazing increases the realism of the simulation since it avoids having agents staring in front of them.

Deliberate vision techniques try to emulate, and even simulate, human vision. The aim of these techniques is to endow the agents with similar sensory capabilities as a real human. Magnenat-Thalmann et al. [Magnenat-Thalmann90] and Noser et al. [Noser95] have developed a synthetic vision system allowing actors to navigate through a corridor using vision, learning and memory. The system works by using a camera to render the scene from the point of view of the agent and then processing the gathered information. The data collected by each agent is constructed individually without accessing the scene database and thus limits the omniscience of the agents. An extension of Noser's work on synthetic vision was introduced by Peters et al. [Peters02]. They used dedicated graphics hardware (Z-buffer algorithm) to efficiently render the scene from the point of view of the agent which provides it with visibility information. False colouring is applied to the objects in the scene dependent on different criteria. This allows the simulation of different vision modes for capturing varying levels of information detail about the environment.

Although vision is the main sense used by virtual autonomous agents to gather information about their environments, some researchers also introduce auditory perception into their models. This enables the virtual humans to perceive auditory events involving objects not in the visual field of view. For example, an agent hearing another agent approaching from behind can choose to turn around and see who is coming. Another effect in modelling aural perception is that some sound events can mask others. A helicopter flying overhead can make it very hard to maintain a conversation at a normal tone of voice and thus prompt the talking agent to shout and even prompt the listening agent to cup its ear to indicate that it cannot hear. In order to create synthetic audition, Kim et al. [Kim05, Kim05b] estimate the sound pressure levels of objects in the environment and compute their individual and cumulative effects on each listener based on the distances and directions of the sources. Another agent that uses sound as well as sight is Flex-Bot [Khoo02], a non player controlled character designed for the first person shooter game Half-Life. Other examples of agents using the hearing sense were developed by Noser et al. [Noser95] and Herrero et al. [Herrero03].

T. Conde proposed a multi-sensory perception [Conde04] that combines information from three distinct sensory systems. For synthetic vision, the same approach described earlier, employed by Noser [Noser95] is used. For synthetic audition, simple cones representing sound propagation are used to represent sound emitting objects. For synthetic touch, a simple collision detection algorithm, similar to V-Collide [Hudson97] is used. These techniques for emulating senses allow an agent to perceive its environment, but do not provide it with any higher level semantic information. It cannot know whether the object it is seeing is a table or a chair, a door that can be opened or an uncrossable wall. Although this type of information can be deduced through top-down, memory and time consuming calculations, it can be also supplied to the agent by the means of direct environment perception, a way of providing the agents with more detailed knowledge in the world.

5.3 Navigating in Complex Environments

One of the most important skills for a Virtual Human is the ability to navigate in a virtual environment, as it is part of a large number of behaviours. Existing 3D Modelling Systems make it possible to generate realistic models of virtual environments, including texture mapping and rendering at multiple levels of detail [Jepson96]. Modelling a complex environment, like a city, with such tools is still a long, complex and costly task. A lot of work has been done to partially automate the rebuilding process. From photogrammetry, Geographical Information Systems, and a Digital Elevation Model, it is possible to construct the model of an object such as a building. For enhanced realism of specific buildings it is possible to combine techniques such as videogrammetry and laser range scanner data [Zhao99] or real-world video and digital maps [Kawasaki99].

To reproduce navigating activity requires additional information beyond the geometric representation of the environment. It is necessary to provide additional data such as mereotopological[3] and semantic information. In his theory of affordance, Gibson states that "animals perceive the environment in terms of what they can do with and in it" [Gibson79]. The "what ... with" aspect has been addressed by P. Becheiraz [Becheiraz98] and M. Kallmann [Kallmann99] with the notion of smart objects. The "in it" aspect has been addressed by N. Farenc [Farenc99] and G. Thomas [Thomas00b] with their respective models of informed urban environments. Each element of the thoroughfares in an urban environment is unique, but it is possible to classify them along a small number of categories [Donikian97]. Concerning pedestrians, it is not possible to restrict the environment to a subcomponent of the city, like the thoroughfares for vehicles, as they can wander about everywhere in the city. However, M. Relieu [Relieu98] explains how a mobile entity uses the urban discrimination to focus its attention, to select pertinent information for its actions inside the current region, while it maintains a secondary task to observe what is happening in regions close to its circulation area.

Based on this fact, G. Thomas et al. have built a mereotopological representation of a virtual urban environment [Thomas00b]. Two complementary data structures are provided for the behavioural entities: hierarchical and topological structures. The components of these structures are different kinds of regions. Three types of regions are distinguished: constrained regions, with a preferred direction of circulation (such as lanes, corridors or sidewalks); intersection regions, where entities can change routes, and free regions, without any constraint on circulation (typically squares). The behaviour is strongly linked to the nature of these regions, as the way an entity circulates depends on the areas it passes through (to wander in a park, to follow a side-walk, to cross a street on a zebra-crossing). In such spaces, entities must also avoid static and dynamic obstacles. The complete city model and the modelling tool they implemented with are described in [Thomas00b]. F. Lamarche has more recently [Lamarche04] developed a new approach, based on the extraction of information from 3D mock-ups. The spatial subdivision algorithm is compounded with several computation steps starting from the 3D geometric database and generating a 2D spatial subdivision using convex cells and identifying bottlenecks (see Figure 7).

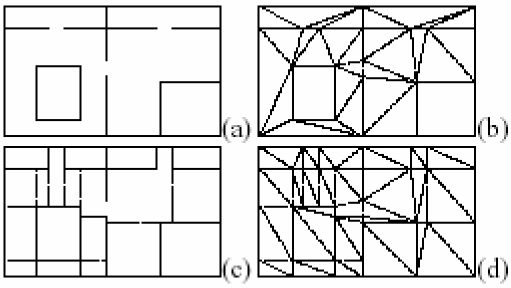

Figure 7: (a) 2D map of the environment. (b) Original constrained Delaunay triangulation. (c) Computed shortest distances between corners and walls. (d) Constrained Delaunay triangulation with shortest distances.

Once the convex cell subdivision is computed, a graph containing topological relations is extracted. A node of this graph is a convex cell and an edge represents a free segment shared by adjacent cells with a length greater than the width of the humanoid. Each node *c* of the graph can be topologically qualified according to the number of connected edges given by

3 Mereology concerns part-whole relationships, while topology concerns connection relationships.

the arity function: closed cell (arity(c) = 0), dead end cell (arity(c) = 1), passage cell (arity(c) = 2) and crossroads cell (arity(c) > 2). This information enables the topological abstraction of the environment: thus, the geometric information related to the low level cells can be omitted and summarized at a more abstract level. The main idea of the abstraction algorithm is to generate an abstraction tree by merging interconnected cells while preserving topological properties, removing dead ends, and simplifying linear paths.

5.4 Path planning

Path planning is the agent behaviour which produces a path in order to reach a specific destination. This process is generally performed by a graph crossing algorithm, which can exploit the hierarchical aspect given by a topological abstraction [Fernandez02]. Path planning and environment representation have been widely studied in the field of robotics, where navigation is a fundamental task to achieve [Latombe91]. In the field of behavioural animation, similar methods are used. Three general approaches can be distinguished: roadmaps [Arikan01], cell decomposition [Thrun96, Bandi98, Kuffner98] and potential fields [Kavraki96]. In [Thomas00] it is proposed to use information about both, the environment circulation areas (risk level) and the virtual human (caution and laziness), to calculate the path. A careless pedestrian tries to take the shortest path, while a cautious pedestrian goes only through spaces dedicated to pedestrian circulation, such as sidewalks and crosswalks.

More recently, R. Thomas and S. Donikian [Thomas03] have proposed to use an IHT-graph structure, filtered in a cognitive map to plan the path. The informed hierarchical-topological graph (IHT-graph) comprises three layers: the Basic Topological Layer containing real urban objects, the Composite Space Layer, and the Local Area Layer. Levels of abstraction allow the agent to plan a route with different granularities. A landmark graph is also used to implement the notion of known paths in the environment. It is especially useful for reactive navigation, low level planning, and re-planning in case an agent should get lost in its environment. The cognitive map “filters” the database access, and the memory model is integrated as part of the cognitive map itself. Each filter object of the cognitive map has a recall parameter and a recognition one.

As some studies show that the recognition and the recall evolution must be considered together [Gillund84], R. Thomas and S. Donikian [Thomas03] have made the decision to consider them embedded in the same structure, with interleaved processes of supply and retrieval encoding. R. Thomas and S. Donikian [Thomas03] have also adapted a model of perception designed by Chopra and Badler [Chopra99, Badler99] introducing three different perception modes depending on the visual attention (exogeneous, endogeneous, and passive). Each time the agent perceives an object, the object memory parameters are modified depending on the type of perception of the agent. Another parameter is used to represent the saliency of the observed object. A complete description of the memory management can be found in [Thomas05, Thomas08].

Thanks to the spatial subdivision presented in the preceding section, it is possible to automatically generate a roadmap enabling path-planning. Inside large and complex environments, this roadmap can also be large and thus reduce the performance of path finding algorithms. However, the common use of the roadmap and the topological abstraction enables a drastic reduction of the path planning graph size, resulting in real-time path finding computation inside large and complex environments for several hundreds of virtual humans. The impact on path finding computation time and complexity is logarithmic (see [Lamarche04] for details and benchmarks). R. Geraerts et al. [Geraerts08] proposed recently the use of the Corridor Map Method for Path Planning to allow online computation of a smooth trajectory for each character based on an offline computation of the graph representing the Corridor Map. The local motion of each character is controlled by using potential fields, which allow to manage obstacle avoidance online. A. Sud et al. [Sud07] propose to use Adaptive Elastic Roadmaps to manage path planning in real-time for several hundreds of characters. An

Adaptive Elastic Roadmaps is a global connectivity graph that deforms based on obstacle motion and inter-agent interaction forces.

Travelled distance has an important impact on path planning, but experimental research has shown that additional factors must be also taken into account [Golledge95]. The least-angle strategy [Hochmair05] minimises the angle between the agent's direction and the destination, allows to take the agent's knowledge about the environment into account, and introduces the notion of preference in the decision process. The simplest path strategy [Duckam03] introduces cognitive cost minimisation for path evaluation. The congestion of an area of the environment [Shao05] is a further important factor, especially for crowd simulation. A stress factor [Osaragi04] can also be taken into account, based on the number of surrounding people, as can the difference between the current and the shortest path. S. Paris et al. [Paris06] base their path planning algorithm on an informed environment description, by using a hierarchical approach with a multi-criteria heuristic. The algorithm is divided into three main steps, one for each level of the graph:

1. Plan in the highest abstraction graph, from the current to the destination zones. If no path is found, then the ending condition cannot be satisfied. Otherwise, proceed to step 2.
2. Plan in the first abstraction sub-graph included in the current zone, and the next one if any. If the zone path contains only a single node, plan from the current to the destination groups, both located in the current zone. Otherwise, plan from the current group to the first group encountered in the next zone. Proceed to step 3.
3. Plan in the informed subdivision sub-graph included in the current group, and the next one if any. If the group path contains only a single node, plan from the current to the destination cells, both located in the current group (the entire path is completed). Otherwise, plan from the current cell to the first cell encountered in the next group (this sub-part of the path is completed and can be used for navigation).

As for any graph crossing algorithm, this hierarchical path planning minimises the cost to find the best path. The cost evaluation method is based on multiple criteria but is mainly composed of two parts: a time to travel cost (based on criteria such as distance, desired speed, and density of population along the path) and a preference cost, including several criteria such as the passage width, the discovery potential, or the deviation direction of intermediate points with origin and destination. In addition, the algorithm only takes into account the connections of the graph which are known by the entity, simply ignoring the others.

5.5 Reactive Navigation

5.5.1 Time-to-collision and personal space

Goffman [Goffman71] describes techniques used by pedestrians to avoid bumping into each other. Avoiding collisions is a spatiotemporal problem. Cognitive science research divided its temporal and spatial dimensions into two different notions: time-to-contact (TTC), and personal space. According to Cutting and colleagues [Cutting95], humans avoid collisions by answering two successive questions:

- Will a collision occur?
- When will this collision occur?

Answers result from the visual perception of one’s environment and of moving or stationary obstacles. Lee [Lee76] and Trésilian [Trésilian91] demonstrated that the optical flow generated from the visual perception of a moving object is sufficient to directly evaluate TTC. The real nature of information used by humans to evaluate TTC is still an open question;

however, humans adapt their motion to avoid collisions in order to preserve admissible TTC. Velocity, distance, and time are intrinsically linked together. As a result, TTC can also be interpreted as a preserved distance between humans and obstacles, giving rise to the notion of personal space. This concept has been primarily introduced by Goffman [Goffman71], who defined it as an oval security region whose front distance corresponds to an *anticipation area* that depends on the pedestrian speed, while the width is the *accepted gap* to pass beside a person or an obstacle or to follow a wall. Goffman also defines the *law of minimal change* that asserts that in his journey, a pedestrian will try to reduce the number and the amplitude of turns. Personal space can be defined as a safety area around them preserved by walkers. The personal space gives walkers sufficient time to react to an unexpected moving obstacle appearing in their perception field. Gérin-Lajoie and colleagues [Gérin05] experimentally measured the personal space's shape and dimensions. They found the personal space to be elliptic, as intuitively imagined by Goffman [Goffman71]. The novelty of this study is to focus on personal space measurement while moving. However, the experimental process was based on the interaction between a human walker and a moving manikin mounted on an overhanging rail.

The social link between strangers is characterized by silence and indifference [Relieu98]. To perform that, different behaviours are used. The first technique called *externalisation* concerns the way people are constantly making others aware of their intentions in order to minimize interactions. Lee et al. [Lee92] show that pedestrians use social conventions such as driving rules to let other people predict their normal trajectory with ease. To selectively gather externalized information from other people, pedestrians use a second technique called *scanning*. The third technique, called *minimisation of adjustment*, expresses that people adjust their trajectories several meters before a conflict to make it perceptible early on by others, with the objective to reduce interaction and avoid coordination needs.

Hillier et al. [Hillier93] show that the majority of human pedestrian movement occurs along lines of sight named *axial lines*. A. Turner et al. [Turner02] propose the EVA system based on a visibility graph. They compare results of this agent-based simulator with real data taken in the Tate Britain Gallery and conclude that they were able to reproduce the aggregate movement with a good correlation. M. Relieu [Relieu98] introduces the notion of *urban discrimination*: pedestrians focus their attention inside their current region to select information of relevance to the activities they are engaged in. A spatial subdivision of the environment is not sufficient to handle navigation, as several moving entities can populate the same environment. In that case, a system allowing dynamic collision avoidance is necessary to achieve consistency and realism.

5.5.2 Reactive approaches

Several reactive approaches can be distinguished, including particle systems [Reich94, Helbing00, Braun03], flocking [Reynolds00, Bayazit02], and behavioural systems [Loscos03]. A particle system assimilates the displacement of an entity to the motion of a particle inside a restricted area. This model, the leading approach in microscopic simulation, is based on physical laws that allow describing attractive and repulsive forces that can be associated to obstacles and moving entities. I. Peschl [Peschl71] justifies the use of this model in the case of an emergency situation with a high population density. When all entities have to move to the same unique exit, a bottleneck occurs and the outgoing flow tends to zero [Helbing00]. Particle based models allow to generate a macroscopically plausible behaviour in case of a high density, but otherwise their failure to take into account perception or social rules becomes apparent. Several evolutions of the original model have been proposed in the last years to expand its domain of application [Braun03, Helbing05, Lakoba05].

Further reactive approaches comprise cellular automata [Schadschneider02] and agent based approaches [Musse99, Lamarche04, Sung04]. More recently, a video based approach has been developed to create a navigation model from examples [Lee07, Lerner07]. However all these approaches are limited to navigation behaviour, omitting management of individual human goals or activities inside the virtual environment.

5.5.3 *Predictive approaches*

Based on experimental studies, J. Pettré et al. [Pettré09] demonstrate that human navigational adaptations are not purely reactive and cannot be modeled exclusively as a function of the distance to static or moving obstacles. It is however possible that this latter case becomes true in the case of crowded areas, where walkers have numerous and intense interactions. Nevertheless, a realistic simulation and animation of virtual walkers in the general case requires modelling of anticipation.

The *steering behaviours* introduced in [Reynolds99] enable interaction solving with anticipation. The unaligned collision avoidance behaviour extrapolates walkers' trajectories – assuming that their velocity is constant – and checks for expected collisions in a near future. A reactive acceleration is computed for each walker, in the directions opposite from the one pointed towards the future collision. Van den Berg and colleagues extend the *reciprocal velocity obstacle* principle from robotics [Berg08]. Similarly to Reynolds' steering, this technique enables collaborative interaction solving with anticipation. Finally, Paris and colleagues [Paris07], inspired by Feurtey [Feurtey00], solve the problem from an egocentric perspective (i.e., walker-centred). In this approach, perceived neighbours' motion is also linearly predicted, and an admissible velocity domain for each walker is deduced. A cost function is used to compute a specific admissible velocity command. More recently, Kapadia and colleagues proposed an egocentric anticipative model in [Kapadia09].

J. Pettré and colleagues [Pettré09] observe that, while during the interaction between two walkers, both make adaptations, it is however the first participant passing who is clearly making less effort: interaction is solved *collaboratively*, but *asymmetrically*. Further analysis also reveals different strategies: the first participant mainly adapts their velocity, whereas the one giving way combines velocity and orientation adaptations. This asymmetry confirms the notion of the personal space: in order to preserve this space, the participant giving way needs to make larger avoidances. Based on 429 experimental samples, Pettré et al. have compared five different algorithms to real data and have made some interesting analyses of the behaviour of each tested algorithm (cf Figure 8).

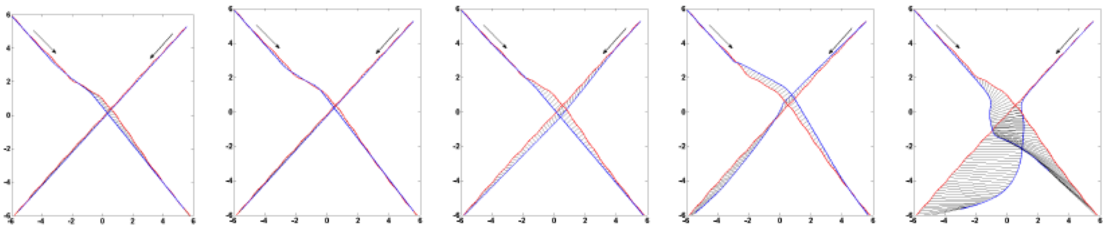

Figure 8: a direct comparison between real interaction (red trajectory) and simulated interaction (blue trajectory) by using five different algorithms: a) Pettré uncalibrated model, b) Pettré calibrated model, c) Reciprocal Velocity Obstacles model, d) Reynolds model e) Helbing model.

Reciprocal Velocity Obstacles (RVO) is an algorithm proposed by van den Berg and colleagues [van den Berg08]; the Reynolds model refers to the open-steer implementation of the steering model proposed in [Reynolds99]; and the Helbing model corresponds to the algorithm described in [Helbing95]. The algorithm that obtained the best likelihood is the calibrated version of the Pettré algorithms, due to its ability to manage both speed and orientation adaptation and to manage the adaptation of the two protagonists of the interaction differently. RVO provides good linear adaptation, but does not use orientation adaptation. The Reynolds algorithm employs both speed and orientation adaptation, but its reaction seems to be too abrupt and is not asymmetric. Finally, in Helbing's model the lack of anticipation is detectable, and the minimal distance between walkers exceeds realistic values.

5.5.4 Collision detection techniques

These techniques differ essentially in the number of simulated entities, level of control, and the associated collision detection method. These types of systems raise the problem of nearest neighbour queries, one of the bottlenecks for the number of possible simulated entities. Several approaches have been proposed to optimize such requests using spatial data structures such as bin-lattice [Reynolds00], K-d trees [O'Hara00] or Kinetic Data Structures [Goldenstein01]. Methods based on the exploitation of an informed environment have been developed [Thomas00b, Farenc99]. This way, specific behaviours related to the type of entity and navigated areas [Hostetler02] have been modelled.

Thanks to path finding algorithms as presented in the preceding section, each virtual human is able to plan its own path. The next step is to follow the path while avoiding collisions with other humanoids and with the environment: The architecture of the reactive navigation model is composed of the computation of the neighbourhood graph (by using a 2D Delaunay triangulation of the humanoid's positions filtered with visibility) and its use in a reactive navigation modular algorithm. The reactive navigation process is described through a pipe filtering the speed vector of the entity [Lamarche04]. The proposed architecture includes a wide variety of pedestrian characteristics and is configurable. It enables the reproduction of a large number of navigation behaviours inspired by psychological studies, such as visual trajectory optimisation and personal space rules, without leaving out real-time constraints. Thanks to the precise representation of the environment through convex cell subdivision, humanoids can navigate in both indoor and outdoor environments.

5.6 Acting on Objects of the World

5.6.1 Introduction

Examples of motion capture interaction can be seen in most of today's video games. A good showcase of such an approach is the game, The Sims [EAgames99]. All possible interactions with all possible objects are recorded using motion capture and then replayed completely unaltered whenever needed. This gives extremely realistic renderings, since it replays motions performed by a live human being. But this realism comes at the price of having the same animation and the same behaviour repeated whenever an action is performed; and this, in the long run, tends to be unrealistic since real humans never perform the same action twice in exactly the same way (cf Figure 9).

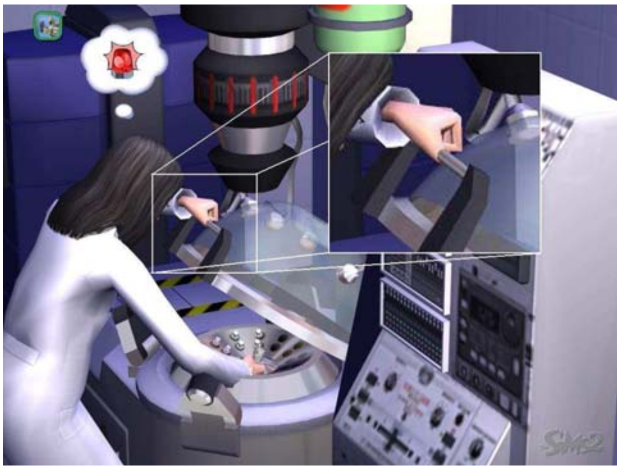

Figure 9: Positioning error during object interaction in the game *The Sims 2* [EAgames04].

5.6.2 PAR: Parameterized Action Representation

A hybrid approach is used by Badler et al. in the PAR system [Badler00] (*Parameterized Action Representation)*, where objects are integral parts in defining the action to be taken. The gestures accomplished by an agent during interaction are completely synthetic, and created by using the EMOTE model [Costa00] which is an interpretation of the Laban Movement Analysis. There are eight principal elements defining a PAR: objects (physical objects concerned by the action), agent (the agent performing the action), application conditions (necessary conditions that should hold to execute the action), preparatory actions (list of condition-action pairs, in which if a condition is not valid, the corresponding action should be executed), action (the action associated to the PAR is either *simple* (execution of a PaT-Net, cf section2.2) or *complex* (list of PAR)), termination conditions (list of conditions specifying when an action is considered to have completed), effects (instructions specifying the update to the world database at the end of the action) and manner (manner in which the action should be performed, including gestures to be accomplished; links PAR to the EMOTE engine).

Thanks to these elements, it is possible to parameterise gestures in a way that affects their outcome, and thus to create differing behaviours adapted to different situations. But the PAR itself is entirely pre-specified and thus needs to be re-adapted offline whenever the targeted object is to be changed. We qualify this approach as "hybrid" because, even though there is no information in the environment itself, the actions to be taken depend directly on the objects they are applied to.

5.6.3 Smart Objects

A more generic approach is taken by Kallmann [Kallmann01] with the Smart Object architecture. A Smart Object, which can hold a relatively complex action, will instruct the synthetic actor about the actions to do step by step. All the information necessary to interact with the object is contained in the object itself. The architecture uses a basic set of primitive gestures which are modified in real time using inverse kinematics to adapt the agent's motion to the position of the object it is interacting with. All the information necessary to interact with the object is contained in the object itself. The smart object instructs the agent about its actions step by step, and all the interaction points are static and defined offline. Tolga Abaci and his colleagues combined Kallmann's approach with a planner to handle long-term decisions [Abaci05]. However, their model offers limited scalability owing to its complexity. The IMPROV system developed at New York University's Media Research Lab integrates procedural animation and behavioural scripting techniques to produce interactive real-time

3D virtual environments [Goldberg97]. In the smart object model variant implemented here, virtual characters get “contaminated” with the skills required to interact or operate an object upon contact. This includes initialisation of a related *skill level* upon first contact and update of this value with each interaction, enabling the portrayal of the character’s gradually improving proficiency.

5.6.4 STARFISH

Badawi et al. have proposed STARFISH [Badawi04], a system allowing the easy definition of interactions between autonomous agents and synoptic objects. Synoptic objects are designed to offer to the autonomous agent a summary, or synopsis, of what interactions they afford. When an agent queries such an object, it learns what actions it can perform, where it should position itself, where it should place its hands, what state the object is in, whether it is allowed to perform the action, etc.. All these indications are given through the use of STARFISH Actions and Interactive Surfaces. STARFISH Actions consist of a group of simple atomic actions, the Basic Actions which are the building blocks used to create Complex Actions. The basic actions are defined as follow:

- TRANSFER, to indicate the agent to move to a new location.
- MOVE, to make the agent move one of its limbs.
- GRASP, to allow an agent to grasp the object. In fact, this instruction acts on the object and makes it follow the limb of the agent handling it.
- INGEST, to put an object into a container. A particular container is the agent's mouth, but the instruction applies to any container in the environment. Some conditions verify that the container is able to receive the object.
- EXPELL, to take the object out of the container.
- TELL, to make an agent or an object emit a piece of information. The action is not limited to a verbal exchange but applies to the transmission of any information detained by the emitter. As an example, an object can be requested to give its current state, and a door would reply that it is open, or closed.
- ATTEND, to wait for the reception of an information, usually as a consequence to a tell request.

These Basic Actions are inspired by the Conceptual Dependency Theory, which seeks to represent complex natural language input using simple basic action verbs. Through the Basic Actions, it is possible to build more complex and varied actions. Figure 10 illustrates the chaining of basic actions used to close a door.

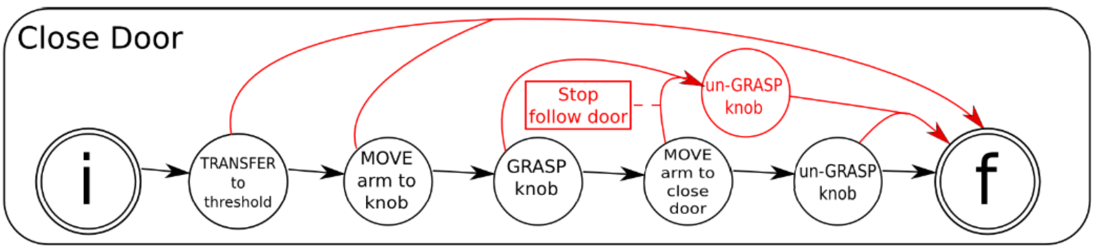

Figure 10: Set of actions instructing on how to close a door [Badawi07].

Interactive surfaces are used to model and describe the affordances of an object. Each object may have as many interactive surfaces as needed, depending on how much interaction it offers. Interactive surfaces are either Interaction Surfaces or Influence Surfaces.

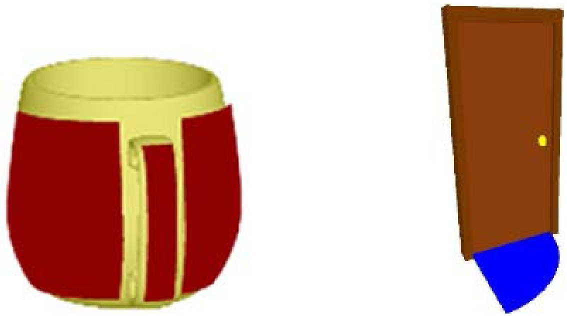

Figure 11: Interaction surface on a cup (dark areas where to place hands to pick it up) and influence surface of a door (area in front of the door to stay clear of) [Badawi07].

Interaction surfaces are used to help the agent handle objects physically. These surfaces are located both on the object and the agent. On the object, they point out interactive parts that are areas on which the agent can perform an interaction. An object's interaction surface can designate a button, as well as areas on a cup where the agent can place its hands to pick it up (Figure 11). The agent uses the influence surfaces to take into account parameters that are not directly part of the interaction, but merely belong to the context of the interaction. For example, Figure 11 shows the influence surface of a door. In this case, the influence is *negative* to prevent the virtual human to be on the door's opening trajectory. Standing out of the trajectory is not a prerequisite to open it, but it will certainly help performing the gesture in a realistic manner, as illustrated in Figure 12.

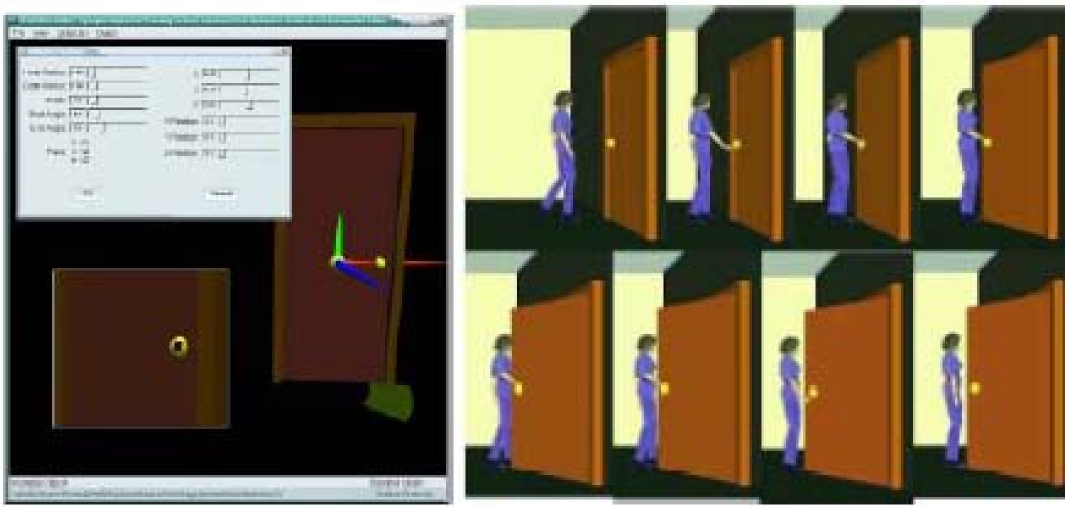

Figure 12: A door modelled in the STARFISH Editor and its use by an agent.

5.6.5 BIIO: Behavioural Interactive and Introspective Objects

Decisions are based on un-embodied notions describing the goals and sub-goals of the agent at a conceptual level. This layer also organises these goals to find feasible actions independent of their location. The logical decision of the autonomous agent is based on concepts describing its goals in the environment. These concepts are unified by an architecture which defines the way they operate together: BIIO which stands for Behavioural Interactive and Introspective Objects.

BIIO manages the concepts as objects in the sense of object oriented languages, i.e., by allowing the creation of a hierarchy of concepts specialising or unifying already existing ones.

Another characteristic of BIIO is that any concept is endowed with introspective abilities, and thus able to communicate its components to others. This architecture also proposes a predefined set of concepts related to interaction behaviours. In BIIO, an interaction is defined as any action between an autonomous agent and another embodied element situated in the environment, such as another autonomous agent or equipment.

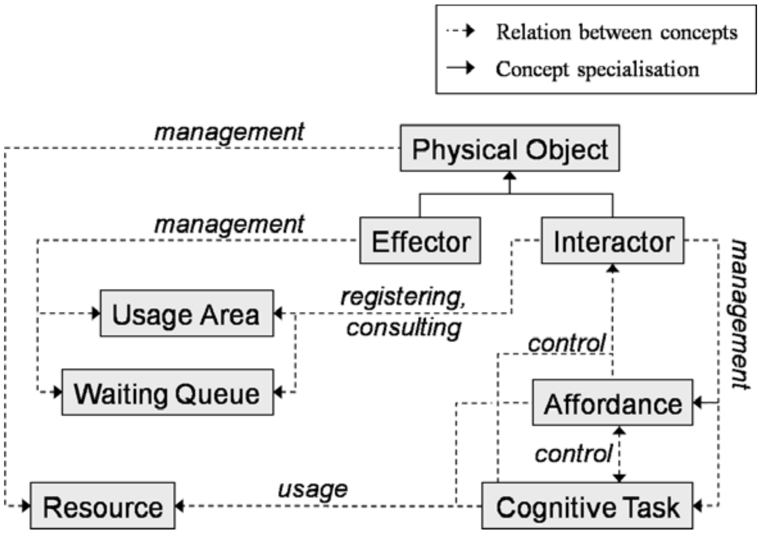

Figure 13: Predefined set of concepts introduced within BIIO.

The set of concepts that are introduced in BIIO are shown in Figure 13. A Physical Object describes a situated object whose location is identified inside the environment. This concept is specialised in two other concepts: Effector and Interactor. An effector represents something which sustains an interaction, i.e. which participates in an interaction without having initiated it. It represents an autonomous agent which can realise an interaction. An interactor is able to manage two concepts representing its behaviours: the Affordance and the Cognitive Task. An affordance describes an interaction between an interactor and an effector, a behaviour of the interactor which is directly observable. These interactions are defined globally in the virtual environment, allowing to easily add new affordances without having to modify neither the effectors nor the interactors. Effectors and interactors are able to collect the affordances they are compatible with, thanks to the introspective abilities provided by BIIO. They further possess generic processes to manage the collected affordances.

A cognitive task describes an internal behaviour of the interactor that is not directly observable. A Resource symbolises a property of a physical object, such as electricity for a piece of equipment, or the ability to move for an autonomous agent. A resource can be *transitory* or *permanent*, and can have multiple *internal properties* which may evolve. Both behavioural concepts (affordance and cognitive task) use resources to define their execution, as for example to manage their competition. A Usage Area describes the positioning of an interactor in order to use an effector. A Waiting Queue describes the way the interactors must socially organise in order to wait for a specific interaction with an effector. Waiting queues are managed in the same way as usage areas. In this approach, the situated decision is managed by a specialisation of the concept of cognitive task, *interact*, which is related to interactions. The goal of this specialised behaviour is to create a connection between the abstract decision of the interactor and its embodied abilities: perception, path planning [Paris06], and navigation [Paris07]. In fact, this cognitive task handles the competition between the interactor's active affordances for the control of the agents physical abilities. *Interact* is specified as a hierarchical state machine, wherein each automaton has a specific role (Figure 14).

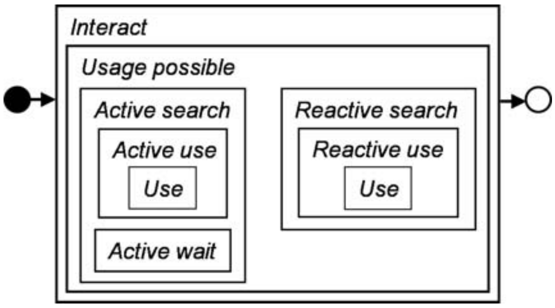

Figure 14: The hierarchical state machine of the BIIO specialised behaviour, *interact.*

The situated decision ability of the agents allows to simulate complex behaviours with a really simple description phase: the author just has to create the pieces of equipment (effectors) with their corresponding affordances, and then to assign an agent its final goal. For example, the behaviour of an outgoing passenger is shown in Figure 15: the final goal assigned to the interactor is to reach a destination by train, producing a set of sub-goals in order to buy a ticket (which creates a ticket resource added to the interactor), check the departure board, and punch the ticket (which modifies a property of the ticket resource). All of these interactions are performed with equipment which is chosen thanks to the interactor's situated cognition ability.

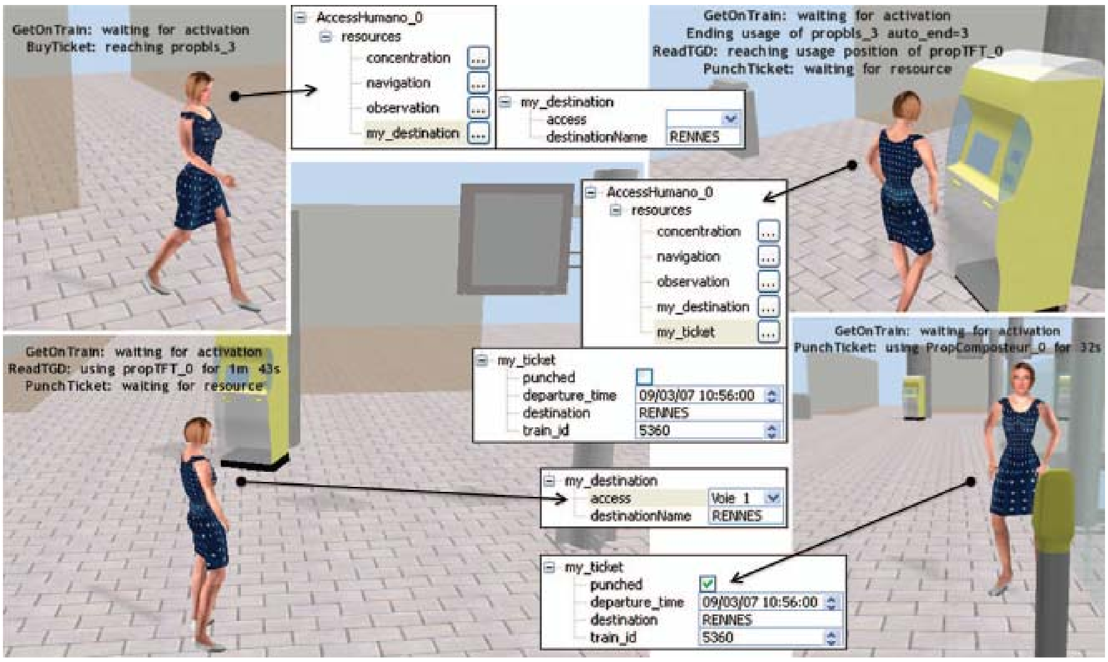

Figure 15: Example of the goal oriented behaviour of an outgoing passenger. The boxes show the interactor's internal states, and the text over the pictures give its affordances.

5.7 Multilayer Approaches

S. Goldenstein et al. [Goldenstein01] have proposed a multi-layer approach to model the behaviour of crowd participants. The first level is a particle system in which each pedestrian is modelled by a particle. The second layer consists of a dynamic topologic structure that maintains the relationship between all moving entities and obstacles by calculating a

Delaunay triangulation at each time-step. The third layer offers a global view that is necessary to plan tasks and determine the desired orientation to reach a dedicated target. Ulicny et al. [Ulicny02] use a layered approach to model the individual behaviour inside a crowd by combining rules and finite state machines. Pelechano et al. [Pelechano06] do the same by combining a rule based approach with the particle system in a Multiagent system called Maces (Multiagent Communication for Evacuation Simulation). More recently, they proposed a new architecture called Carosa (Crowds with Aleatoric, Reactive, Opportunistic, and Scheduled Actions) which is a composite architecture dedicated to functional crowds built on Maces [Pelechano08]. It also uses PAR to manage character actions. This approach employs a scheduler to select an action based on role and object definitions.

Q. Yu and D. Terzopoulos introduced hierarchical decision networks to model virtual humans' action selection in the presence of uncertain knowledge [Yu07]. Their model is based on an earlier one that connects the cognitive and reactive layers and adds memory abilities to the agents [Shao05b]. In their research, cognitive decisions are based on purely logical considerations and do not account for an action's space and time requirements. Thus, even though this model is a breakthrough toward multilayered behaviours, the set of addressable behaviours is still restricted by the lack of interdependency between logical and situated decisions.

Paris et al. [Paris08] propose a multilayered behavioural model (Figure 16) that includes four interconnected layers: biomechanical, reactive, cognitive and rational. Each layer has it own frequency and exchanges specific information with the directly superior layer to inform it of some imposed constraints, and also with the directly inferior layer to control it. The rational layer is in charge of the logical decisions of the agent based on its own objectives and its spatial cognitive map. Coupled with a generic management of functional objects and cognitive tasks, this creates a connection between the abstract decisions of the agent and its embodied abilities: perception, path planning, and navigation.

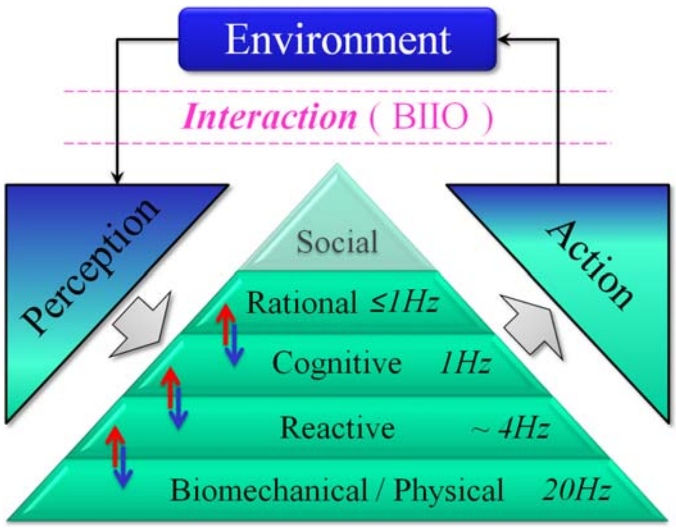

Figure 16: A behavioural model for activity-driven populace, with multilayered decisions. The red (upward-pointing) and blue (downward-pointing) arrows show how information propagates between layers. For each level, the typical refresh rates are shown.

6. Theories Proposed in Cognitive Science

6.1 Introduction

The hierarchical nature of levels of behaviour is generally recognised today. For A. Berthoz [Berthoz06b], the complex problem of the motor control was solved in nature by a hierarchy of levels of control where each is applied on internal models of the system which precedes it[4]. The theory of control in behavioural psychology as described by R. Lord and P. Levy [Lord94], takes up the principle of force-feedback loops, while spreading it to all behavioural processes, from the simple task of visual servoing to the regulation of social behaviour. For Lord and Levy, the generality of retroaction loops for the description of behaviour results from the hierarchical nature of the systems of control, even if the nature of the activities of control can be very different according to the levels. The common point between all these levels lies in the comparison between a perceived state and an expected state and in maintaining the error within acceptable limits. Every level knows only the immediately lower level and receives only errors already elaborated by internal models which compare a wished state to a real state.

The brain contains several schemas of the body (mechanisms of simulation of action) that are independent from the real body itself [Berthoz06b]. The superior organs which take decisions do not necessarily work by arranging sensory information directly. These centres know only the state of the inferior levels of execution which contain models of the levels which they control and in particular which estimate the errors between what they imposed and what is executed.

Lord and Lévy [Lord94] stated the hypothesis that control of the human processes is produced by a mutual interaction between two mechanisms: *top-down* and *bottom-up*. Top-down control is abstract and strongly connected to the current intentions and to established plans, whereas the bottom-up control is more guided by the data, so conveying the information supplied by the perceptive system to recognize the conflicts. The ascending regulation is a necessary complement to the downward control mechanism, to assure that the cognitive system will note and answer correctly to physiological and psychological demands. Nevertheless, the detection of conflicts at the hierarchical level corresponding to the management of tasks should not create too much additional cognitive load to that resulting from symbolic reasoning. On the other hand, conflicts between upstream and downstream processing are assumed to be capable of interrupting thought to redirect the attention and to generate new models capable of surmounting them.

Such hierarchical nature of behaviour is taken into account in the models of type hierarchical parallel state machines with a management of bidirectional exchanges between levels (downward control followed by ascending information feedback). The models of type hierarchical ASM and PHISH-Nets also integrate a hierarchical structuring, but with an ascending mechanism of communication only that allows the superior level to choose among the actions proposed at the lower level.

---

4 Even so, we do acknowledge the existence of “deep connections” e.g. in the human nervous system, which connect higher level directly to the lowest ones and to sensors and effectors.

6.2 Memory

6.2.1 Introduction

In psychology and cognitive science human memory has been studied for a very long time, and hence, many models and approaches currently exist. However, the notion of modelling human memory has been much more scarcely tackled in computer science: a few models are widely used, but they rely on a small number of concepts, usually borrowed from the social sciences. Nevertheless, interesting and practical models exist whatever the followed path: if inspiration from cognitive psychology leads to memory models that are close to the human functioning, computer science proposes innovative solutions of its own, which sometimes provide a better fit to the needs and possibilities of computer simulation.

One of the first general theories of memory was the modal model of memory suggested by Atkinson and Shiffrin in 1968 [Atkinson68]. For them, the memory consists of several stages and there are not one but three kinds of memories: sensory registers, short-term memory (STM, also called working memory) and long-term memory (LTM). The sensory registers correspond to the five senses: touch, taste, smell, vision and audition. STM is transitory and of limited capacity while LTM is more or less permanent and has a very substantial potential capacity.

6.2.2 Working Memory

In 1974, Baddeley and Hitch [Baddeley74] showed that short-term memory is actively committed to cognitive processes and that it should not be regarded as having a secondary role of throughway, but rather as an active working memory storing and handling information for complex cognitive tasks. They proposed a model of working memory built around three components (Figure 17): The central executive, the phonological loop and the visuospatial sketchpad.

Figure 17: The working memory model as defined by Baddeley and Hitch in 1974 [Baddeley74].

The phonological loop is related to the treatment of speech-based information. It stores auditory data for a short time and is believed to be involved in many processes related to auditory input: organisation and rehearsal of articulatory data, combination of information from both the auditory sensory input and the central executive, but also the storing of visual information that have articulatory features. According to Baddeley, this component of the model has evolved with findings in psychology and is certainly the most developed of the three.

The visuospatial sketchpad processes and temporarily stores visuospatial information. It handles visual and spatial information separately and is able to combine them. The central executive controls attention and merges information stored in the two subsidiary short-term storage systems. A further development of the model strengthens the links between the short-term components and the corresponding long term components [Baddeley88]. In this model, the latter also communicate with an episodic memory[5] that combines information from the various long term memories.

[5] A more detailed description of episodic memory as part of long term memory is developed in section 6.2.3.

Both storage systems within the model (phonological loop and visuospatial sketchpad) are seen as slave systems in charge of temporary storage of visual and spatial information. The central processor is the most important element of this architecture [Baddeley90]. It is in charge of the attentional mechanism of action selection described by Shallice [Shallice82] (section 6.3).

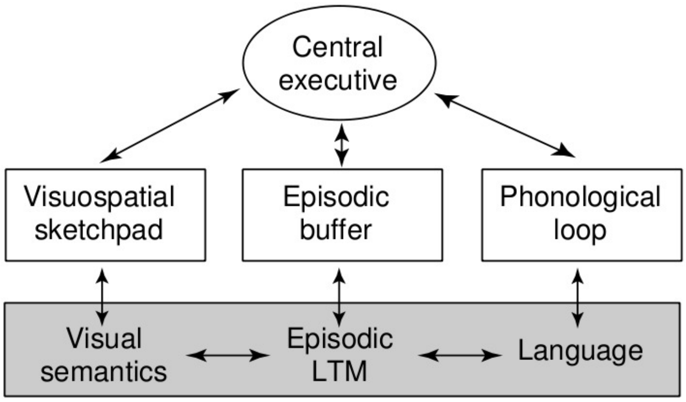

Figure 18: The updated working memory model published in [Baddeley00].

In [Baddeley00] an episodic buffer is introduced in working memory (Figure 18) to explain the close integration of working memory's information with events in time and space. Still, this episodic buffer does not replace the episodic memory in LTM: its insertion only intends to help the combination of data working memory from both the central executive and the episodic long term memory.

6.2.3 Long Term Memory

Semantic Memory

Semantic and episodic memories make up together the category of declarative memory, which is one of the two major divisions in memory. The counterpart to declarative, or explicit, memory is procedural, or implicit, memory (see also below). Semantic memory includes generalised knowledge that does not involve memory of a specific event. For instance, you can answer a question like "Are wrenches pets or tools?" without remembering any specific event in which you learned that wrenches are tools.

Semantic memory is the portion of long term memory which is concerned with ideas, meanings, and concepts which are not related to personal experiences [Collins69]. Together with episodic memory, it makes up the section of the long term memory known as declarative memory. Long term memory also includes procedural memory, which is the memory of how to do things (see also 6.2.4). These three different kinds of long term memory all interact with each other to allow people to do everything from reading a book to flying a space shuttle.

Semantic memory can also become confused during the early stages of learning, as for example when someone struggles to understand that two radically different styles of chair are both considered chairs, while also grasping the difference between a chair and a bench. Semantic memory is especially active in childhood, as children constantly encounter new concepts which must be defined and filed away.

Semantic memory plays a key role in many human activities. For example, procedural memory provides information about how to read a newspaper, but it is semantic memory which remembers what the different letters mean, and how they link together into words. Semantic memory also allows a reader to understand written communications in multiple fonts, since the brain understands the concept of a letter, rather than a specific example of a letter.

Episodic Memory

Episodic memory refers to the memory of events, times, places, associated emotions, and other conception-based knowledge in relation to an experience. [Tulving83] introduces it as the memory of a person's experiences with both temporal and spatial relations to the experiences themselves.

C. Brom et al. [Brom08b] integrate an episodic memory module into a virtual agent design to perform tasks (in that version, this was still limited to search for objects) in a virtual environment. Brom's episodic memory model consists of a long term memory module that stores and forgets information according to a task, objects used or needed during the task, a dating and an emotional salience. The memory is organised around a hierarchy of tasks. It also links each task with the relevant resources in the scene. Dating and emotional salience are used in the forgetting process, to decide what is important to keep and what should be deleted.

In order to search for objects in the environment, the LTEM (Long Term Episodic Memory) is linked to a spatial memory module. Other parts of the architecture include visual treatment of the scene, modelling of emotions, task selection, and conflict resolution for the preparation of the action. The simulation was run in a virtual environment which the size of a house for scenarios lasting a couple of days, and the virtual human behaved correctly. Authors now plan to test the behaviour in a simulation which the size of a city, with the same kind of *search* task [Brom08].

An interesting feature of the model is its organization of the memory around tasks, which serve as a backbone. The action is therefore put at the centre of the model, instead of the description, as in more classical models.

6.2.4 Typologies of memory

As already mentioned, some authors distinguish between explicit and implicit memories. Explicit Memory refers to a memory that is conscious of passed events, revealed by mechanisms of recall and of recognition, while the implicit memory refers just to the results of the experiment, thought, or action of a person which are ascribable to an event passed, independently of any conscious memory of this event. Kihlstrom [Kihlstrom99] enumerates various experiments, in particular related to amnesia, which tend to corroborate the parallel and independent existence of these two kinds of memory.

The most popular theories in cognitive neurosciences postulate that explicit and implicit memories are completely dissociated. Tulving and Schacter [Tulving90] proposed that implicit memory is based on several systems of perceptual representation localized in various parts of the cortex which store representations specific to the method concerned, but not the meaning of the stimulus. Squire et al. [Squire95] postulate that explicit memory is localized in and near the hippocampus.

Another distinction introduced in the literature regards direct and indirect memory, in reference to whether the studied items are present or not at the time of access to the memory. Thus, mechanisms of recall and recognition are inevitably explicit and direct, because on the part of the subject they require a mechanism of recall conscious of items studied previously.

Table 1 presents typologies of memories according to Kihlstrom [Kihlstrom99].

Memory	Explicit	Implicit
Direct	Recall Recognition	Completion of a radical Completion starting from fragments
Indirect	Proactive Inhibition Retroactive Inhibition	Free Association Generation of Categories

Table 1: Memory Types.

R. Sun et al. [Sun05] also introduce a distinction between implicit memory and explicit memory in the CLARION cognitive architecture. The distinction concerns both declarative and the procedural knowledge. Sun's work on implicit and explicit memory concerns the modelling of related differences and complementary aspects of both kinds. Pointing out that experiments show that people are able to learn some reasoning tasks without understanding explicitly the underlying logic [Berry88], Sun's model includes implicit procedural knowledge and implicit declarative knowledge as well as explicit knowledge. This model of implicit memory uses neural networks, while explicit memory is rule-based and chunks-based. Indeed, Sun states that implicit knowledge is not *consciously* perceived by the system, contrarily to explicit knowledge. According to Sun, neural networks and rule-based planners respectively integrate these features. We describe the CLARION architecture in section 7.4, and in particular we detail the components of the system and the role of implicit and explicit memory modelling.

Squire et al. [Squire84] distinguish procedural and declarative long-term memory[6]. The declarative memory corresponds to the notion of knowledge and consists of storage of knowledge about the world within a network of connections, whereas the procedural memory represents know-how and stores knowledge about how to do things. Declarative knowledge is factual by nature and can be represented as propositions while procedural knowledge relates to mental or behavioural operations and can be represented as a condition-action production rule system. It is moreover postulated that declarative knowledge is available for conscious introspection while procedural knowledge is unconscious. Conscious knowledge concerns only percepts, images, and thoughts that come to mind when unconscious procedural knowledge operates on declarative knowledge that is itself unconscious.

Other theories postulate on the contrary that there is only a single memory system, and that any perceived differences do not lie in physical separation but in the types of requests to the same memory system [Mandler80] or in the multiplicity of the modes of representation used to think of or perceive events when they occur [Engle99].

6.3 Attentional Mechanisms for Action Selection

Shallice [Shallice82] has proposed two kinds of attentional mechanisms for the selection of actions: the *Supervisory Attentional System* and the *Contention Scheduling System*. The *Contention Scheduling System* is used to select in an automatic manner actions to be done when a situation is routine. The typical example is car driving: when past the learning phase, a driver can change speed and orientation while talking with passengers or listening to the radio. To be able to do that, action selection should occur automatically, as an important part of conscious cognitive load is used for speech information processing.

The driver can shift gear, which implies a simultaneous manipulation of the clutch pedal and the control lever, while conversing with the other passengers, listening to the radio, or thinking of scheduling their activities of the week-end. When confronted with a new situation

[6] The ACT-R model presented in section 7.3 is also based on this separation.

or when parameters of a situation enforce a change of habit, the *Contention Scheduling* cannot intervene any more, because it cannot select the adequate actions itself. In that case, it is the *Supervisory Attentional System* which forces choices triggered by *Contention Scheduling*. The *Supervisory Attentional System* is a system with a limited capacity, used for a variety of intentions, including:

- Tasks involving planning and decision-making;
- Situations in which the automatic reaction process appears to get in trouble;
- New situations;
- Technically difficult or dangerous situations.

Out of all the models presented previously, only HPTS++ can manage this coordination of several threads of activity in an automatic way, thanks to its management of physical resources.

6.4 Mechanisms of Activation and Inhibition

The brain is a simulator of action, a generator of hypotheses. Anticipation and prediction of consequences of actions based on the memory of past ones is one of its fundamental capabilities. There is neither any mechanism of perception apart from action, nor any mechanism of attention apart from action selection. To decide is not only to choose between several solutions, it consists also in blocking undesired behaviours, i.e. inhibiting [Berthoz06b]. Processing of tasks is protected from interruptions, particularly for complex tasks (e.g., [Kuhl84, Halperin95]). Inhibiting mechanisms are very important as they prevent unnecessary elements to enter the working memory[7]. A mechanism of will is used to avoid any interruption of an ongoing behaviour through a direct inhibition of all competitive behaviours (e.g., [Lord94, Blumberg94]. R. Lord and P. Levy [Lord94] suggest that several complementary rules are at work:

- **Proposition 1:** the instantiation of a goal is going to privilege the closest information in terms of category:
  - (a) by increasing the speed with which this information can be reached;
  - (b) by increasing the importance to access such information.
- **Proposition 2:** the activation of a goal is going to prevent the instantiation of competitor goals:
  - (a) by increasing the latency of their activation;
  - (b) by reducing the priority to access such information;
  - (c) by producing negative primary effects.
- **Proposition 3:** the normal realization of an intention deactivates the referring structures, by releasing the cognitive system of the positive and negative affects.
- **Proposition 4:** the repeated failure of a goal can deactivate the referring structures, by releasing the cognitive system of the positive and negative affects.
- **Proposition 5:** the automatic follow-up and detection of conflicts is an important bottom-up control mechanism which integrates biological needs and processes at the symbolic level.

---

[7] Active memory dealing with both the treatment and preservation of requested short term information

Deliberation implies the existence of at least two candidate solutions. The brain has to make a clear choice between competing solutions. A. Berthoz [Berthoz06b] postulates that a double mechanism of modularity and hierarchy is used:

- Modularity, because the basal ganglia of the thalamocortical base are specialized in the control of the movements of the glance, of the gesture or of the memory.
- Hierarchy, because it is possible that these parallel modules have a hierarchical crosswise connection

A. Berthoz [Berthoz06b] presents three kinds of architectures that have been proposed in the literature for action selection:

1. Subsumption: actions endowed with a hierarchical index, allowing to select them automatically with respect to a fixed order.
2. Organization of actions in a network: mutual inhibitive connections between all actions in a distributed action network that is connected to sensors and effectors.
3. Supervisor or central selector: it activates the circuits selectively. This approach allows to drastically reduce the number of connections and offers at the same time more flexibility. It combines the advantages of modularity and centralism.

A link can be made between the first approach and the behaviour rules and finite automaton approaches presented in section 2. The second approach can be mapped to the sensor-actuator approach also presented in section 2 and by the competitive action selection mechanisms presented in section 3. As far as we know, the only representative of the third approach is HPTS++.

6.5 Activity Theory

For W.J. Clancey [Clancey02], activity theory is a precursory form of situated cognition that will be presented in the next paragraph. The three levels of the activity theory are:

- the activity (motivations): forces operating on the decision taking process;
- the action (goals): what should be done;
- operations (conditions): how it should be done.

M. Sierhuis, in his doctoral dissertation [Sierhuis01], defines activity as follows:

> *An activity is a collection of actions performed by one individual, socially constructed, situated in the physical world, taking time, effort, and application of knowledge. An activity has a well-defined beginning and end, but can be interrupted.*

To understand the notion of activity, it is necessary to take into account the fundamentally social nature of human activities. Our activities as human beings are always forged, forced and made significant by our continuing interactions with the worlds of work, the family, and the other communities to which we belong. An activity thus is not only something we do, but a way of interacting. Any human activity is deliberated, but a purpose is not necessarily a problem that must be resolved, and not every action must be necessarily motivated by a task to be carried out. W.J. Clancey [Clancey97] takes the example of a person who listens to music while driving their car on their way back home. This activity is a part of the driving practice for many persons, but is in no way a necessary sub-goal to reach their destination.

According to Clancey, the motives underlying human behaviour are imperfectly characterized through the problem solving theory introduced by Allan Newell in his unified theory of cognition [Newell90]. Not all goal oriented behaviours are obtained by inference or compilation. Certain actions simply reproduce cultural motives, whereas some others are coordinated without deliberation by using mechanisms of attention and adaptation. Clancey discusses the notion of parallelism of tasks. A person does not perform multitasking in parallel, rather, several tasks take place simultaneously by merging attentively several parallel interests. Clancey states that the conceptualization of this notion is still primitive and that its nature is underdeveloped in neuropsychological theories.

In [Berthoz06], A. Berthoz and J.L. Petit assert the existence of at least five closed loops of neuronal circuits (basal thalamus-cortex-ganglions) which work in an autonomous way and in parallel, thereby allowing to control the movements of eyes, limbs, memory, and feelings: these closed loop systems coordinate in a dynamic way. Contrary to the problem solving approach, Clancey defends that an activity is not necessarily interrupted when another need arises, as, for example, when another activity becomes pressing or if an external condition comes to interrupt what the person was doing. This would imply a mechanism of competitive activation. The starting and ending of an activity are more subtle than a purely goal oriented decision. Clancey states that there is no opposition between notions of sequencing and parallelism, but rather a coupling of them. These coupled sub-systems need to be organised in real-time. Parallelism allows to organise several activities at the same time, whereas serialism forces the processing of ordered forms of sequences of action. Parallelism is fundamental to couple behaviours and to temporally order them, in particular when they regard several sensori-motor modalities.

Out of all behavioural models, only the hierarchical parallel state-machines allow this combination of the sequential and parallel aspects, and within these models only HPTS ++ appears to be able to take into account the various sensori-motor modalities and the activation/inhibition principles in a generic way.

6.6 Embodied and Situated Cognition

Representation is the central notion of traditional cognitive science to the detriment of the physical and environmental context in which cognitive systems are brought to operate. It is common to consider that only the symbolic approach can be used to model cognitive behaviours, because these were often conceived as problems to be solved in the form of condition-action rules. However, this traditional approach of cognitive science and artificial intelligence has been confronted with challenges such as the Chinese room [Searle80]: the fact of disposing of symbols and rules allowing to manipulate them does not yet define any kind of intelligence, because at no time there is usage of their meaning.

In reaction to such criticism, the notions of situatedness and embodiment gained in importance in cognitive science at the end of 1980s [Ziemke02]. In this perspective, cognition is not reduced to an intellectual demonstration any more, but it also and importantly involves the body and the physical (and social) environment(s). The body of the agent as well as the environment in which it operates induce structures which the internal cognitive devices of the cognitive agent can and need to use to solve the agent’s problems [Agre97,Keijzer02].

Intelligence can then be described as the sum of the previous physical experiences of the agent acquired through its interaction with its environment. In other words, the intelligence of an agent is based on its previous interactions with the physical world. Brooks [Brooks90] introduced the hypothesis of the importance of physical grounding which postulates that the intelligence of an agent must be based on the interaction between the physical agent and its environment. According to Harnad [Harnad90], symbols have to acquire their common sense, it is what he calls the symbol grounding problem:

> *An artificial system based completely on the manipulation of symbols never glimpses semantics which is associated with them.*

To find a solution to this problem, he proposes that symbols be based on a process of invariance extraction from sensori-motor signals, defined in three stages:

- Iconisation: the processing of the signals into iconic representations or icons;
- Discrimination: the capacity to judge whether two inputs are identical or different and if they are different, in what way they differ;
- Identification: the capacity to assign a unique answer to a class of inputs, treating them quite as equivalents in a certain way.

Whereas there is a consensus in the situated and embodied cognition community in critiquing the traditional approach to cognition as computational process as incomplete or even erroneous, there is, on the other hand, no clear consensus on the definition of the foundations of the new approach, as reflected in the recent proposals of interactionism or enactivism (e.g. [Bickhard95,Vernon07,Adolphs06]).

6.7 Ecological Theory

W. Warren [Warren95] stated that “the goal of perception is not to provide the general description of a scene but to extract specific information for the task implied in the activity in progress.” and thus joined J. Gibson [Gibson79] who proposed an ecological approach to perception, a theory which associates a behaviour-based semantics with each object of the environment. J. Gibson developed a theory of the affordances, a term that is not easily to translate, but which essentially corresponds to the perceptible physical availabilities and invitations of an object, place, or situation. The stress is laid not on the nature of observed but on the nature of the observer who wants to access immediately the characteristics of the object or the environment which interests them.

Thus, an environment will provide directly to the behavioural entity all of the possible behaviours with respect to the objects and entities which constitute it. Gibson defines a typology of affordances: media, substances, surfaces and their provision, objects, other people and animals, places. This theory postulates that it is more useful to know in advance the nature of the interactions with an object than to have precise concepts of its geometry and to attempt recover its characteristics from that information.

The affordances introduced by J. Gibson are opportunities for action that an object, a place or an event provide for the cognitive agent. They refer to the resources encountered by the cognitive agent in its environment. This concept of affordance is not dissociable of that of capacity or aptitude (or as yet others say, effectivity). For example, the affordance of a chair is related to our capacity to fold the basin and the knees to be seated. The aptitude is a means of acting that a cognitive agent can use to carry out a particular affordance. The perception of affordances was demonstrated to be dependent on the body of the observer and the latter is a variable notion because depend on whether or not tools are being used (e.g., the cane of a blind man transforms the range of affordances). According to Hirose, the borders of the body are dissociated by the processes of perception and of action and so the use of a tool modifies the perception-action loop [Hirose02].

Before use, the tool is an independent object, separated from the body of the cognitive agent. It has specific affordances and gets opportunities of action. During use, a tool is not an object any more; it acts as a functional extension of the agent. It plays a central role by extending or transforming the aptitudes of the agent to identify and carry out affordances in its environment. When a tool extends the capacities of an agent, it also extends its body.

Gibson was criticized to have based his theory only on perception while neglecting the cognitive process.

6.8 Conclusion

The main finding regarding the investigation on existing links between action selection models and work in cognitive science on action selection mechanisms is that there are very few relations. There are only few discussions in the behavioural animation literature or in artificial intelligence about the cognitivist foundations of the proposed models, apart from the recent field of the embodied and situated cognition. This last research field is really multidisciplinary and is gathering researchers from cognitive science, cognitive robotics and artificial intelligence. Finally, within the reactive models proposed in computer science HPTS++ appears to be the sole model that integrates most of the presented notions.

However, decision making is not self-sufficient and should be integrated in a multilayered behavioural architecture. Usually, such architectures are based on rule-based systems and refer to classical disembodied and unsituated cognitivism . New cognitive architectures are being developed that take the perspectives of interaction, embodiment and situation, and enactivism into account, e.g. in EC-funded artificial cognitive systems research, with the umbrella coordination actions euCognition and euCog II (see e.g. [Vernon07]). .

7. Cognitive Architectures

7.1 Introduction

Cognitive architectures aim at unifying within the same model broad sets of properties collectively ascribed within the scientific community to human cognition. We are now going to present some prominent main models of cognitive software architectures that attempt to support modelling of human behaviour.

7.2 Soar

One of the most ambitious efforts to unify various aspects of cognition is the Soar [Laird03, Laird08, Laird09] system, rooted in Allan Newell's work on a unified theory of cognition [Newell90]. Soar can be described as a symbolic architecture for intelligence that integrates basic mechanisms for problem solving, the use of knowledge, learning, and sensori-motor behaviours. Soar provides a unique architecture for all tasks and sub-tasks, a single representation of permanent and temporary knowledge, a mechanism to generate goals and a learning mechanism.

In Soar, all decisions are based on the interpretation of perceived data, the contents of working memory created for the solution of previous problems, and any knowledge derived from permanent memory. Soar is based on the hypothesis that all deliberated goal oriented behaviours can be represented by the selection and application of operators to states. A state is the representation of the current situation of a problem to be solved that is currently in memory. The application of an operator changes the current state into a new state, by changing its representation in memory. A goal is a desired outcome of the problem solving activity. Soar continuously applies operators chosen out of the currently applicable ones based on their perceived potential to contribute to achieving the goal currently pursued.

Soar possesses two types of memory: working memory and long-term memory. The working memory contains descriptions of the current situation, including the data obtained from sensors. The working memory is organized in objects described through their attributes. The current state is the sum of all objects contained in the working memory with the current values of their attributes. The long-term memory (called also production memory) contains productions, i.e., conditions-actions rules. If the conditions of a production hold within the current state contained in the short-term memory, then actions associated with the production are candidates for execution. Soar embodies different notions of learning: learning by experience, which supports the generation of new production rules, and, as more recently, reinforcement learning. Coordination in Soar is modelled in the form of three types of planning:

- extended: abstraction and selection of goals;
- hierarchical: selection of operations to reach goals that are in accordance with a plan calculated in advance;
- reactive: selection of motor actions.

The learning step moves some knowledge from the extended planning level to the intermediate level of deliberated actions and also to the lower level of reflexes. Soar is used in several applications in various fields. In particular, it is used at the Institute of Creative

Technologies (University of Southern California) in their Virtual Human Architecture to model cognitive agents [Hartholt08].

7.3 ACT-R

ACT-R (Automatic Control of Thought) [Anderson04, Taatgen06] is a cognitive architecture aiming like SOAR to propose a coherent vision of human cognition. In contrast to SOAR, the architecture of ACT-R is modular and every module is dedicated to the processing of a specific type of information. Behaviour coordination occurs within a central production system. This central processor has no direct access to the data of modules and can only access a minimum of information put in the buffers of the various modules. So an individual will not be receptive to the whole information being in their field of view but only to that of the objects to which their are attentive. In the same way, they are not immediately conscious of all the contents of their long-term memory, but only of recovered facts. The Figure 19 below presents the global architecture of ACT-R.

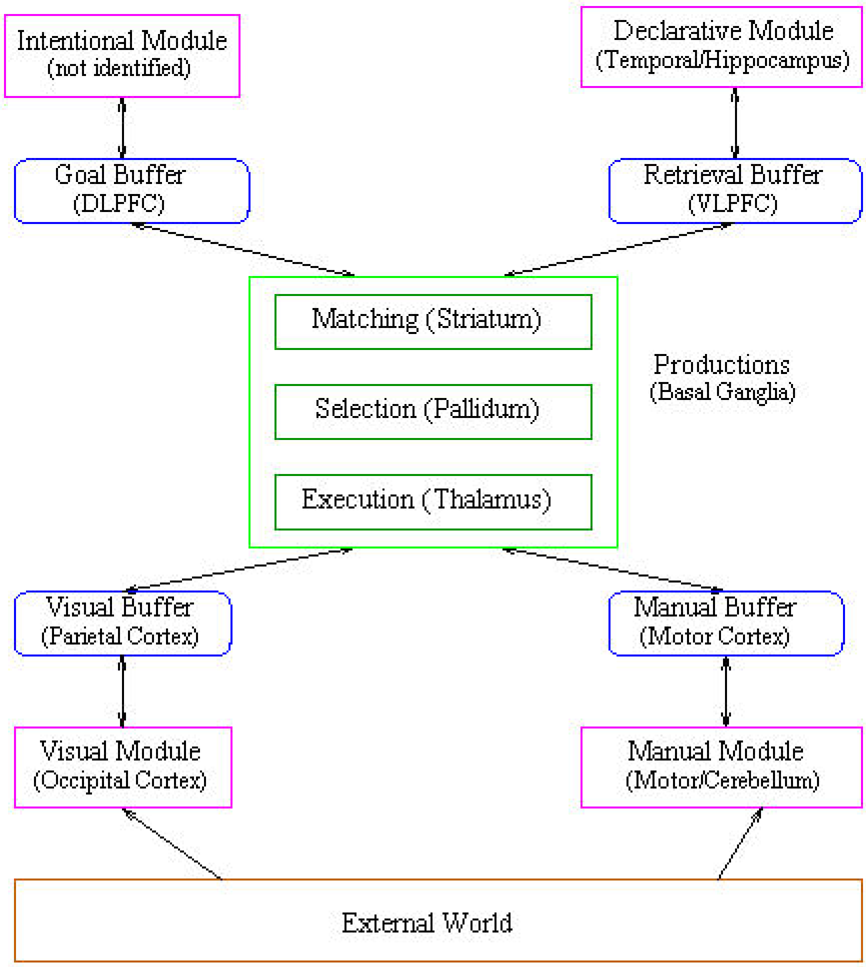

Figure 19: The organization of information in ACT–R.

In this figure, we can notice an association between modules and cortical regions in which specific processes are supposed to take place . For example, DLPFC stands for DorsoLateral PreFrontal Cortex and VLPFC for VentroLateral PreFrontal Cortex. The design of ACT-R also distinguishes two types of long-term memory: declarative memory and procedural memory. The architecture supposes a mixture between processes that are executed in sequence and in parallel. Inside any module, there is a strong parallelism. For example, the visual system treats the whole field of view simultaneously and the declarative system executes parallel searches through memory in answer to requests to recover information. Processes executed within the various modules are executed in parallel in an asynchronous way and there are only two bottlenecks in the system. The first one is that the content of a buffer is limited to a single unit of knowledge, called "chunk" in ACT-R. In this way, only a single element can be recovered from long-term memory at each cycle and in the same way only a single object can be e.g. encoded out of the full field of view. The second bottleneck is that only a single production can be selected at each cycle, a substantial difference from Soar.

A Hybrid cognitive-reactive architecture has been proposed by M.D. Bugajska et al. [Bugajska02] by coupling ACT-R with SAMUEL, an evolutionary algorithm-based rule-learning system. SAMUEL, by using genetic algorithms, provides capabilities to learn simple condition-action rules. The two components communicate by message passing. In the presented example, SAMUEL is used to model the individual collision free navigation behaviour of a Micro Air Vehicle, while ACT-R is used to model team behaviour.

7.4 CLARION

CLARION (Connectionist Learning with Adaptive Rule Induction ON-line) is a cognitive architecture that incorporates the distinction between implicit and explicit memory (Figure 20) [Sun03, Sun05]. It is composed of four subsystems:

- the action-centred subsystem (ACS), whose role is to control actions;
- the non-action-centred subsystem (NACS), whose role is to maintain general knowledge;
- the motivational subsystem (MS), whose role is to provide underlying motivations for perception, action, and cognition;
- the meta-cognitive subsystem (MCS), whose role is to monitor, direct, and modify the operations of all other subsystems.

Implicit and explicit representation of data is part of each subsystem.

The Action-Centred Subsystem takes decisions related to actions of the agent within the world. To select the action that will be performed, the implicit and explicit parts of the ACS first take their own decisions separately, according to their respective knowledge. Then the ACS chooses which action is to be performed. Eventually, the memory of both representations is updated according to changes in the environment induced by the performed action. Sun notices that the dual nature of the decision process enables his model to compute and learn in both a top-down and bottom-up fashion. The former is achieved when a rule is successfully applied: the neural network learns under the supervision of the rules. When an action selected by the implicit process succeeds, the system is capable of creating rules to enhance its ontology. [Browne01] reviews rule-based connectionist inference models and [Apolloni02] addresses the extraction of rules from a dataset using a connectionist network.

The Non-Action Centred Subsystem describes the general knowledge about the environment, i.e., the knowledge that does not take part in the ACS decision process. Linked to the ACS, the NACS subsystem stores and represents data as an associative memory (implicit and distributed encoding) and a set of associative rules (explicit, local representation of connected chunks). The Motivational Subsystem and the Meta-Cognitive Subsystem represent the supervisory processes of the system, complementing the ACS and NACS which are operating subsystems. The MS introduces an account of context in CLARION. It manages drives explaining and influencing decisions taken by the ACS, and, by extension, knowledge stored by the NACS. The MCS monitors, controls and regulates the ACS and the NACS.

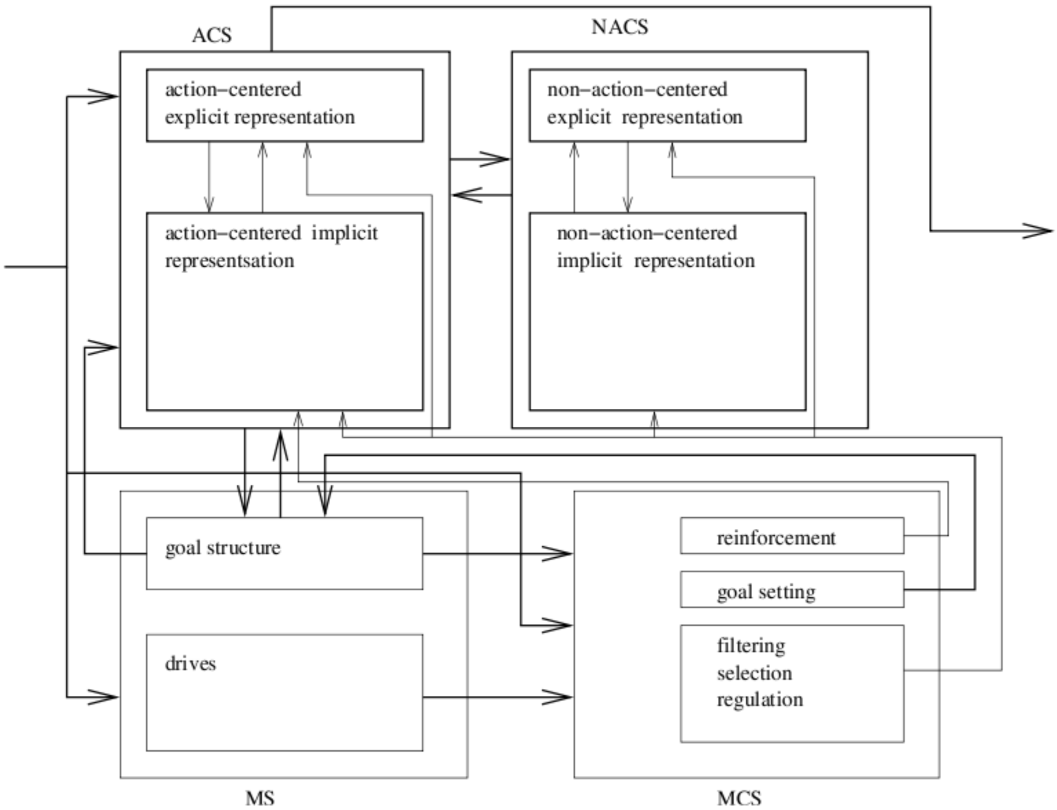

Figure 20: The CLARION architecture [Sun05].

7.5 EPIC

EPIC (Executive-Process/Interactive Control) is a cognitive architecture developed by David E. Kieras and D. E. Meyer at the University of Michigan. The objective of EPIC is to provide a precise simulation of human data processing, in particular with a detailed description of the perceptive, cognitive, and motor activities (**Error! Reference source not found.**). An important characteristic of EPIC is its orientation towards the simulation of situations involving the management of multiple tasks.

EPIC implements a synthesis between, on one hand, a production system representing knowledge processing and, on the other hand, the simulation of parallel systems of data processing by the various sensori-motor circuits. EPIC admits two categories of parameters: the standard parameters that can be considered as parameters of the stable system in all tasks, and specific parameters that are free to vary across situations, but usually still observe conventional value ranges. An example of a standard parameter in EPIC is the duration of a production cycle in the cognitive processor, with a value of 50 ms. An example of specific parameter is the time that the visual processor takes to recognize that a particular shape represents a straight arrow, with a value of 250 ms.

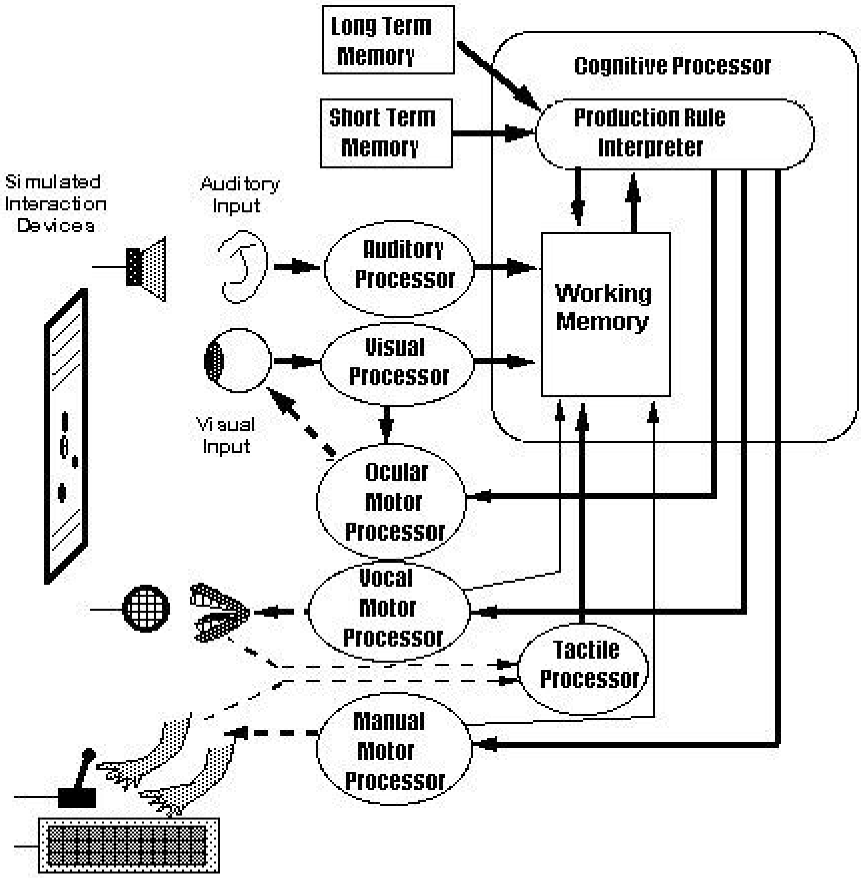

Figure 21: The EPIC Architecture.

However, EPIC does not simulate the sensory and motor processes as such; it rather simulates the temporal scheduling and the durations, by taking into account constraints of coordination and sharing of resources between the various steps of the sensory and motor processing. For example, concerning the simulation of the realisation of gestures and actions, the preparation and realisation phases are distinguished. In EPIC, only a single movement can be in preparation at the same time, and only a single action can be in progress at any time [Byrne03].

The time to prepare a movement depends on the number of determining elements of the movement which must be prepared for every movement, and the characteristics of the last prepared movement. What was prepared for the previous movement can sometimes be reused, which saves time. EPIC can thus realise quickly the repetition of equivalent movements, because the laborious preparation of characteristics is not necessary if successive movements are identical. If they are not, the time gained is a function of the scale of the difference from the previous movement. The execution time for a movement corresponds approximately to the time necessary for its physical realisation; the execution time for voluntary movements of hands and fingers is calculated according to Fitts' Law.

Next to the knowledge about the task, an additional set of rules, called executive knowledge, is also mobilized in the simulation. This covers reporting of results: when more than one task is realised at the same time, they can run in parallel. However, numerous sensori-motor processors are serial: People have e.g. only a single pair of eyes which can be managed only towards a single place at once; thus, if multiple tasks are in progress and require the observation of different areas, something has to arbitrate. In EPIC, this additional knowledge about the way to manage multiple tasks is called execution rules (executive knowledge), and the productions implementing this knowledge run in parallel with the memory of productions implementing the task knowledge.

7.6 CoLiDeS

CoLiDeS stands for Comprehension-based Linked model of Deliberate Search and it has the objective to model the way users explore a new interface. Contrary to the previous systems, it is not based on production rules, but on a model of construction–integration developed by W. Kintsch [Kintsch98]. CoLiDes is based on a process whose components and logic differ largely from the cycles of production systems. Each cycle includes two different phases: the construction phase and the integration phase. Properties of both phases can be summarised as follows: in the construction phase, an initial input (e.g. the contents of the current display device) is introduced into a weakly constrained rule-based process which produces a proposition network.

The items of the network are connected on the basis of overlapping arguments. Once the construction phase is finished, the system is left with a proposition network. In the following integration phase, activation propagates through the network in the style of a neural network. This phase is essentially a phase of constraints satisfaction used to select a proposition of the network as the favourite proposition.

Another difference lies in the degree of knowledge with which the model is initially endowed. While in the production systems it is the knowledge that characterises the task to be realized, CoLiDeS models a minimal amount of knowledge associated to a minimal number of instructions. So, an important contribution of the model is to simulate the way the user instantiates appropriate goals by trying to bring the task defined in the instructions to a successful conclusion. Mechanisms of attention regulation as well as the integration of LSA (Latent Semantic Analysis) were also included [Landauer97].

7.7 Conclusion

There are a number of similarities between ACT-R and Soar. The production rules of the procedural memory of ACT-R correspond to the production rules in the long-term memory of Soar. The facts contained in the declarative memory of ACT-R correspond to the objects in the working memory of Soar. The goals of ACT-R correspond to the operators of Soar. In both Soar and ACT-R, deliberation is set and separated by immediate perception / action coordination. Deliberation is seen as a system state taking up a number of cycles at least in between perception and movement, independently of the current activity.

Compared with SOAR and ACT-R, the production system of EPIC presents at least two important differences: on one hand, there is no limitation of the number of production rules which can be started during a cycle of the cognitive processor and, on the other hand, the level of granularity of the rules is much higher, authorizing parallelism at every level of the architecture, including the cognitive level. Therefore, EPIC is strongly directed towards the elaboration of predictive models of the performance of subjects having to manage several tasks in parallel. A more detailed comparison of cognitive architectures can be found in [Taatgen08].

Clancey criticises that deliberation is not a kind of time-out for action. Deliberation in fact occurs as a sensori-motor experience and does not take place before or between perception and the action. Deliberation is not a higher level process in the sense of control but in the sense (direction) of the organization of the way we perceive (collect), order, or give meaning to material and experiences (experiments) already created.

8. Existing Software Components on Virtual Humans developed by IRIS Partners

8.1 Introduction

The four software components described below have been developed over many years by members of the Bunraku Research team, which is project-team of INRIA jointly with INSA of Rennes, ENS Cachan, CNRS and University of Rennes 1. The software components have been licensed to the Golaem Company. Golaem is a French spin off company of BUNRAKU, whose main objective is to develop software solutions for autonomous human-like characters able to perform realistic and complex actions in real time. An agreement between the owners of the software components and the company allows the Bunraku team to provide free licences for partners involved in collaborative projects such as IRIS; however, the use is restricted to the purposes of the collaborative research project, without any commercial use.

8.2 MKM, also known as Golaem Motion

MKM (Manageable Kinematic Motion) is a real time animation engine for synthetic humans. Based on kinematics, it automatically blends, mixes, and adapts actions to different morphologies and to constraints. The use of priorities for all actions allows an intuitive control of the animation. MKM's animation loop is based on a modular pipeline structure which permits to easily build animations adapted to specific needs and hardware resources: less complicated and less constrained animation for crowd simulation, or full precision animation with fewer characters.

Usually, animations are created by animators in a 3D modeller (e.g., 3DSMax, Maya) directly on the desired virtual character. Consequently, the resulting motions can hardly be animated on characters with different morphology. On the opposite, MKM stores motions using a morphology-independent representation of the skeleton. This representation allows the animation of hundreds of different characters on common computers. Each action is associated to intrinsic constraints (such as ensuring foot contact without sliding, reaching targets,...). MKM offers an authoring tool called MotionMaker in order to edit these constraints, ensuring that they will be verified in the real-time animation engine.

In order to avoid blending of incompatible motions (such as a left hopping motion and a locomotion with the right foot on the ground), synchronisation is necessary. This process takes the sequence of stances into account in order to ensure in real-time that a motion is compatible with all the running ones. If two motions are not compatible, it applies dynamic time warping in order to solve the problem. Ground adaptation is based on the calculation of footprints depending on the current actions. In real-time, these footprints are corrected in order to take the shape on the ground into account.

MKM offers an automatic motion blending algorithm based on the computation of weights according to priorities intuitively and interactively tuned by the user. Priorities are associated to each action and to each body group. For example, walking is associated to high priorities on the legs while grasping uses high priority for the active arm. This allows to grasp an object while walking.

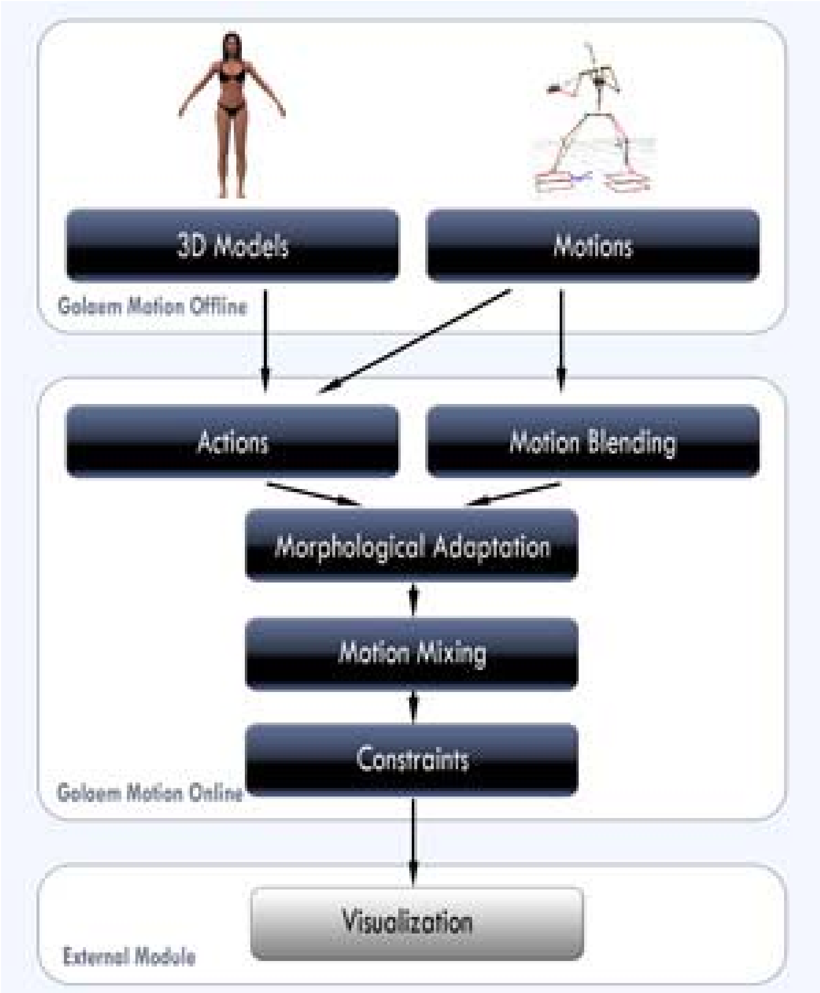

Figure 22: MKM Workflow.

MKM is predictable and is able to compute as many future footprints as wanted. These can then be modified in any way through several different levels: from the lowest one, which allows to modify the position and orientation of any footprint (put the feet in a realistic configuration when climbing stairs for example), to the highest one, which offers the possibility to define a global direction for the motion (to reach a specific point, for example).

A real-time 3D inverse kinematics solver is included in MKM in order to adapt the motion to additional constraints. The engine also provides a centre of mass controller to ensure that the mixing of several motions always respect the basic rules of dynamics.

8.3 HPTS++, also known as Golaem Coordinator

HPTS++ is a platform independent toolkit, that allows to describe and handle the execution of multi-agent systems. It provides a specific object oriented language, encapsulating C++ code, for interfacing facilities and a runtime kernel providing automatic synchronization and adaptation facilities.

HPTS++ is composed of a language allowing agent description through finite state machines and a runtime environment handling parallel state machine execution and offering synchronization facilities. It has been used in different research fields such as behavioural animation, scenario description and automatic cinematography. Its scheduling system provides new paradigms for multi-agent systems description while ensuring the overall consistency of the execution.

The language provides functionalities to describe state machines (states and transitions) and to inform them with user specific C++ code to call at a given point during execution. It is object oriented: state machines can inherit of other state machines and/or C++ classes to provide easy interfacing facilities. States and transition can be redefined in the inheritance hierarchy and the state machines can be augmented with new states and transitions. Moreover, state machines are objects that can provide a C++ interface (constructor / destructor / methods) for external calls.

The task model provides a framework dedicated to the description of behaviours through tasks and operators described through HPTS++ state machines. Thanks to this model, it is possible to describe primitive behaviours through atomic tasks and combine them, thanks to the provided operators, to rapidly and easily create complex behaviours. Provided tasks operators are sequence (on success then, on failure then), parallelism, loops (while, for), alternative, all without order. Those operators are fully dynamic. Hence, they can be used at runtime to dynamically describe complex behaviours.

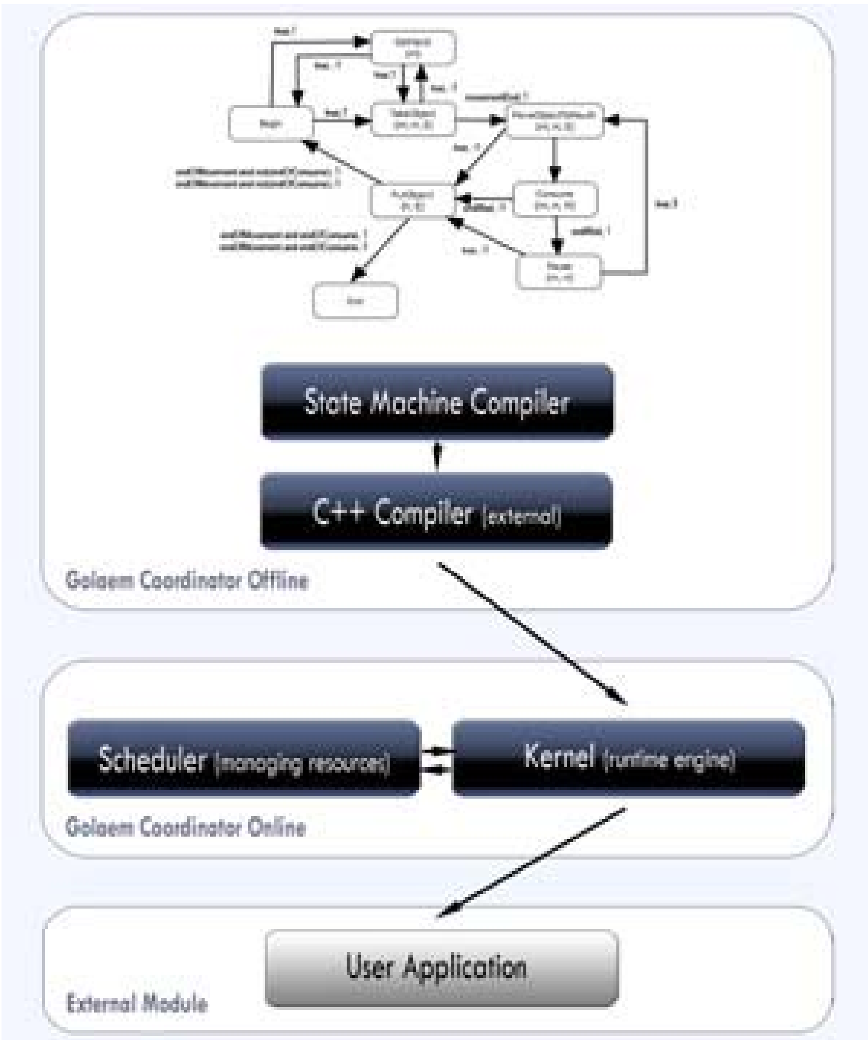

Figure 23: HPTS++ Workflow.

8.4 TopoPlan, also known as Golaem Path

TopoPlan is a toolkit dedicated to the analysis of a 3D environment geometry, in order to generate suitable data structures for path finding and navigation. This toolkit provides a two step process: an off-line computation of spatial representation and a library providing on-line processes dedicated to path planning, environmental requests...

TopoPlan is based on an exact 3D spatial subdivision which accurately identifies floor and ceiling constraints for each point of the environment. Thanks to this spatial subdivision and some humanoid characteristics, an environment topology is computed. This topology accurately identifies navigable zones by connecting 3D cells of the spatial subdivision. Based on this topology several maps representing the environment are extracted. Those maps identify obstacles and step borders as well as bottlenecks. Based on this representation, several concise and accurate roadmaps are generated to handle real time path planning within the environment. This spatial representation is computed off-line thanks to a tool provided with the library.

Provided algorithms and data structure do not relate to a virtual human animation toolkit. They are stand alone and can be easily connected to user architecture. Based on the computed roadmaps, TopoPlan provides a 3D path planning algorithm enabling to plan path within very constrained environments. The path generated by the path planning algorithm may not be realistic. Thus, TopoPlan provides a trajectory optimization algorithm enabling the generation of shorter and more realistic trajectories. In order to accurately control a virtual human navigating inside a 3D environment, a footprint generation process is provided. This process can be coupled with the trajectory optimization algorithm to generate footprints along a trajectory that respect environmental constraints such as correctly putting a foot on a step of a stair for example.

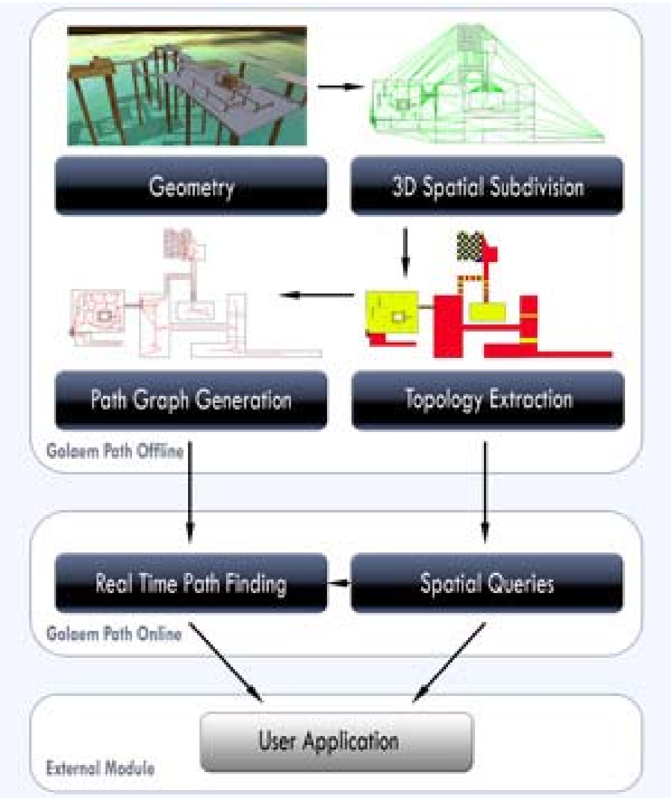

Figure 24: TopoPlan Workflow.

8.5 Yalta, also known as Golaem Activity

Yalta (Yet Another Language for Task Analysis) is a set of tools designed for human activity description and simulation. Yalta consists of:

- a graph based language dedicated to human activity description;
- an authoring tool for designing human activities using this language;
- a runtime engine processing the activities described with this language.

These tools can be used as a pipeline from an activity description to a runtime simulation. The Yalta Language enables to describe and formalize any activity: it can be simple behaviours as well as complex collaborative tasks. Complex activity structures can be designed by applying several connectives such as alternatives, parallel processing, events synchronization... It relies on a hierarchical activity model which enables to specify a description at different levels of detail. This property aims to use the same description while speaking with someone about the global workflow of an activity or tuning the details of a simulation.

Yalta also consists of a runtime engine encapsulating HPTS++ and thus, provides the same advantages (hierarchical parallel state machines system, automatic resources synchronization…). This engine can be easily integrated in any C++ architecture. Yalta provides an offline generator responsible for parsing a Yalta Language file (containing activity descriptions) and building a C++ library. Then, this library can be run by the Yalta Engine to perform a simulation.

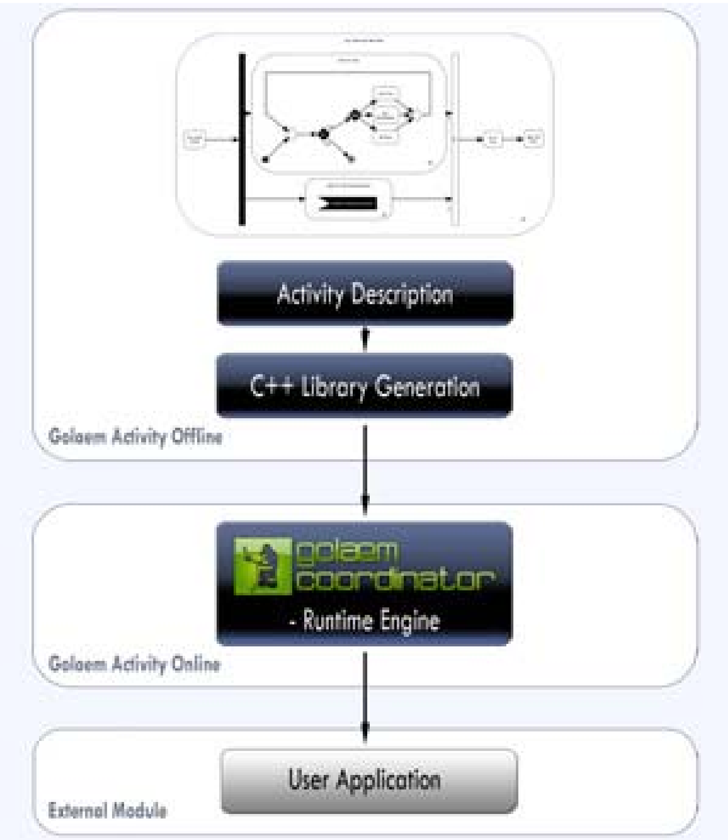

Figure 25: Yalta Workflow.

9. Requirement Specification

What is needed for autonomous virtual actors in interactive storytelling is an open architecture of virtual humans able both to perform autonomously actions that are not important for the dramaturgy and to follow requests by the drama manager (planner) which are mainly dramatic actions to perform. Of course, a feedback from the actor to the drama manager is needed to monitor the evolution of execution of actions, and to be able to replan if an action fails or a goal state becomes unreachable. Another important element is the link with the virtual cinematographer, as it may be important to control the location or displacement of characters to make sure that they will be shot correctly, without any obstacle in-between the camera and the characters. Two ways can be envisaged:

1. to control precisely the location of characters in the environment based on predefined camera placement or possible locations;
2. to dynamically locate the camera based on cinematographic idioms and to impose locally displacement constraints to the characters during each sequence.

The first approach requires the precomputation of possible trajectories of characters based on the camera locations, while the second one imposes to be able to dynamically find the best location for cameras, based on the local path of each actor (difficult to estimate in advance when using reactive navigation with obstacle avoidance).

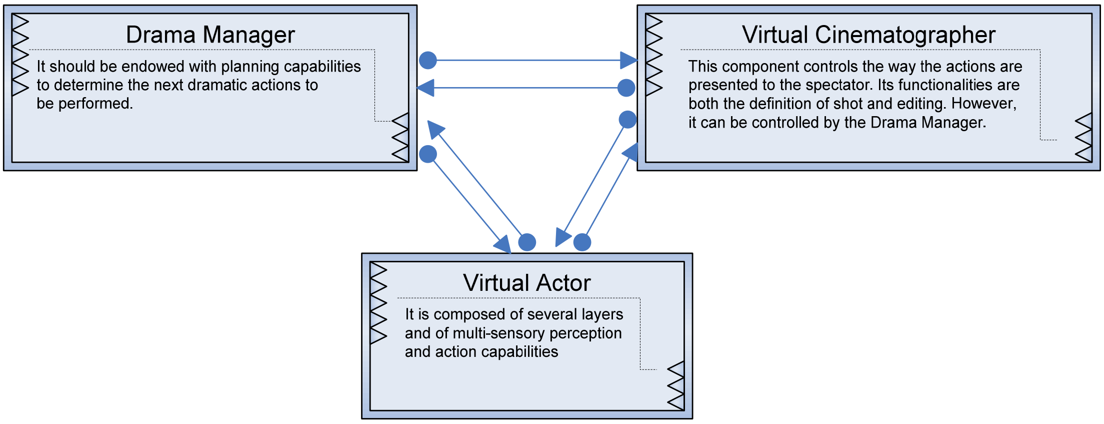

Figure 26: Links between the different modules of an Interactive Story.

10. Conclusion

Simulating natural-looking virtual humans requires modelling different behavioural levels to mimic how people choose and organise their activities. A multilayer behaviour model for virtual actors can help interactive story developers endow each entity with high-level objectives provided by the drama manager and build on top of a reactive and cognitive decision making system. The model should ideally integrate a decision process that provides bidirectional links between the different layers. Each layer informs the layer directly above it of specific information regarding imposed constraints and controls the layer directly beneath it. Each layer is built independently and exchanges only a set of identified data. In this way, bottom-up and top-down approaches can be combined allowing the character to behave autonomously when there are no narrative issues, and it can be fully or partially controlled when requested by the drama manager and/or the virtual cinematographer. However, even though partial virtual human architectures have been proposed in the past, such an ideal architecture is still to be developed.

11. References

[Abaci05] T. Abaci, J. Ciger, D. Thalmann, *Planning with Smart Objects*, Proc. of Int'l Conf. Computer Graphics, Visualization and Computer Vision (WSCG 05), 2005, pp. 25–28; http://wscg.zcu.cz/wscg2005/Papers_2005/Short/D19-full.pdf.

[Acquisti03] A. Acquisti, W.J. Clancey, R. van Hoof, M. Scott, M. Sierhuis. BRAHMS Tutorial, Version 1.0, July 2003.

[Adolphs06] Adolphs R.: How do we know the minds of others? Domain-specificity, simulation, and enactive social cognition, Brain Research, 1079(1):25-35, 2006.

[Agre97] P. Agre, I. Horswill: Lifeworld Analysis, *Journal of Artificial Intelligence Research*, 6:111-145, 1997.

[Ahmad94] O. Ahmad, J. Cremer, S. Hansen, J. Kearney, P. Willemsen. *Hierarchical, concurrent state machines for behavior modeling and scenario control*. In Conference on AI, Planning, and Simulation in High Autonomy Systems, Gainesville, Florida, USA, 1994.

[Anderson76] J.R. Anderson. *Language, memory, and thought*. Lawrence Erlbaum Ass., Hillsdale, NJ, 1976.

[Anderson04] J.R. Anderson, D. Bothell, M.D. Byrne, S. Douglass, C. Lebiere, Y. Qin. An integrated theory of the mind. *Psychological Review*, Vol. 11, N°4, pp 1036-1060, 2004.

[Apolloni02] B. Apolloni, D. Malchiodi, C. Orovas, G. Palmas. From synapses to rules. *Cognitive Systems Research*, 3(2):167-201, 2002.

[Arikan01] O. Arikan, S Chenney, D. Forsyth. Efficient multi-agent path planning. In *Computer Animation and Simulation '01* (2001), Springer-Verlag, pp. 151–162.

[Atkinson68] R.C. Atkinson, R.M. Shiffrin. Human memory: A proposed system and its control processes. In *The Psychology of Learning and Motivation*. pp. 89–95. – New York: Academic Press, 1968.

[Badawi04] M. Badawi and S. Donikian. *Autonomous Agents Interacting With Their Virtual Environment Through Synoptic Objects*. CASA'04, the Computer Animation and Social Agent Conference, Geneva, July 2004.

[Badawi07] M. Badawi, S. Donikian. The generic description and management of interaction between autonomous agents and objects in an informed virtual environment. *Computer Animation and Virtual Worlds*, 18(4-5):559-569, 2007.

[Baddeley74] A. D. Baddeley, G.J. Hitch. Working memory. The *Psychology of Learning and Motivation: Advances in research and theory*, 8:47-89, 1974.

[Baddeley88] A. D. Baddeley, C. Papagno, G. Vallar. (1988). When long-term learning depends on short-term storage. *Journal of Memory and Language*, 27(5):586-595, 1988.

[Baddeley90] A.D. Baddeley. *Human Memory: Theory and practice*. – Lawrence Erlbaum Associates, 1990.

[Baddeley00] A. D. Baddeley. The episodic buffer: a new component of working memory? *Trends in cognitive sciences*, 4(11):417-423, 2000.

[Badler97] N.I. Badler, B.D. Reich, B.L. Webber. *Towards personalities for animated agents with reactive and planning behaviors*. Lecture Notes in Artificial Intelligence, Creating Personalities for synthetic actors, (1195):43-57, 1997.

[Badler99] N. Badler, D. Chi, S. Chopra. Virtual human animation based on movement observation and cognitive behavior models. In *Computer Animation'99*, pp. 128–137, Geneva, Switzerland, May 1999. IEEE.

[Badler00] N.I. Badler, R. Bindiganavale, J. Allbeck, W. Schuler, L. Zhao, S.J. Lee, H. Shin, M. Palmer. *Parameterized action representation and natural language instructions for dynamic behavior modification of embodied agents*. In AAAI Spring Symposium, 2000.

[Bandi98] S. Bandi, D. Thalmann. Space discretization for efficient human navigation. *Computer Graphics Forum 17*, 3 (1998).

[Bayazit02] O.B. Bayazit, J.M. Lien, N.M. Amato. Roadmap-based flocking for complex environments. In *Pacific Conference on Computer Graphics and Applications* (2002).

[Becheiraz98] P. Becheiraz, D. Thalmann. A behavioral animation system for autonomous actors personified by emotions. In *First Workshop on Embodied Conversational Characters (WECC '98)*, Lake Tahoe, California, 1998.

[Berg08] J. van den Berg, S. Patil, J. Sewall, D. Manocha, M. Lin. Interactive navigation of individual agents in crowded environments. *Symposium on Interactive 3D Graphics and Games (I3D 2008)* (2008).

[Berry88] D. Berry, D. Broadbent. Interactive tasks and the implicit-explicit distinction. *British journal of psychology*, 79(2):251-272, 1988.

[Berthoz06] A. Berthoz. *Emotion and Reason. The cognitive neuroscience of decision making*. Oxford University Press, 2006.

[Berthoz06b] A. Berthoz, J.L. Petit. *Physiologie de l'action et Phénoménologie*. Odile Jacob, 2006.

[Bickhard95] Bickhard M.H., Terveen L.: Foundational Issues in Artificial Intelligence and Cognitive Science, Elsevier, Amsterdam/New York, 1995.

[Blumberg94] B. Blumberg: Action-Selection in Hamsterdam: Lessons from Ethology*, Proc. of 3rd International Conference on the Simulation of Adaptive Behavior*, Brighton, England, pp. 108-117. MIT Press, 1994.

[Booth93] M. Booth, J. Cremer, J. Kearney. *Scenario control for real-time driving simulation*. In Fourth Eurographics Workshop on Animation and Simulation, pp. 103-119, Politechnical University of Catalonia, September 1993.

[Braun03] A. Braun, S.R. Musse, L. De Oliveira, B. Bodmann. Modeling individual behaviors in crowd simulation. In *16th International Conference on Computer Animation and Social Agents* (2003), IEEE.

[Bratman87] M.E. Bratman. Intentions, Plans, and Practical Reason. Harvard University Press: Cambridge, MA, 1987.

[Bratman88] M.E. Bratman, D.J.Israel, M.E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349-355, 1988

[Brom08] C. Brom, T. Korenko, J. Lukavsky. *How Place and Objects Combine? Long-term Spatial Memory Enhanced by Objects for Human-like Agents*. Technical report, Department of Software and Computer Science Education, Charles University in Prague, 2008.

[Brom08b] C. Brom, J. Lukavsky. *Episodic Memory for Human-like Agents and Human-like Agents for Episodic Memory.* In AAAI, editor, Biologically Inspired Cognitive Architectures: Papers from the 2008 AAAI Fall Symposium, Arlington, Virginia. Association for the Advancement of Artificial Intelligence, 2008.

[Brooks85] R.A. Brooks R.A.: *A Robust Layered Control System for a Mobile Robot*, AI Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1985.

[Brooks90] R.A. Brooks. *Elephants don't play chess*. Robotics and Autonomous Systems, 6:3-15, 1990.

[Browne01] A. Browne, R. Sun. Connectionist inference models. *Neural Networks*, 14(10):1331-1355, 2001.

[Bugajska02] M.D. Bugajska, A.C. Schultz, J.G. Trafton, M. Taylor, F.E. Mintz. *A hybrid cognitive-reactive multi-agent controller*. Proceedings of the IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, EPFL, Lausanne, Switzerland, October 2002.

[Byrne03] M.D. Byrne. *Cognititive architecture*. In The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, L. Erlbaum, Mahwah, NJ, 97-117, 2003.

[Chopra99] S. Chopra-Khullar, N. Badler. Where to look ? automating attending behaviors of virtual human characters, in *Proceedings of Autonomous Agents '99,* Seattle, WA, (1999) ACM.

[Clancey97] W. J. Clancey. *Situated Cognition: On Human Knowledge and Computer Representations*. Cambridge University Press, 1997.

[Clancey98] W.J. Clancey, P. Sachs, M. Sierhuis, R. van Hoof. Brahms: Simulating practice for work systems design. *International Journal on Human-Computer Studies* 49: 831-865, 1998.

[Clancey02] W. J. Clancey. *Simulating activities: Relating motives, deliberation, and attentive coordination*. Cognitive Systems Research, 3(3):471-499, 2002.

[Coderre88] B. Coderre. *Modeling behavior in petworld*. In artificial life, pp. 407-420. Addison-Wesley, 1988.

[Collins69] A.M. Collins, M.R. Quillian. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8, 240-247, 1969.

[Conde04] T. Conde, D. Thalmann. An artificial life environment for autonomous virtual agents with multi-sensorial and multi-perceptive features. *Computer Animation & Virtual Worlds*, 15(3-4):311-318, July 2004.

[Costa00] M. Costa, L. Zhao, D. Chi, N.I. Badler. *The EMOTE model for effort and shape*. In SIGGRAPH, 2000.

[Courty03] N. Courty, E. Marchand. *Visual perception based on salient features*. In IEEE/RSJ International Conference on Robots and Systems (IROS), Las Vegas, Nevada, 2003.

[Cutting95] J. Cutting, P. Vishton, P. Braren. How we avoid collisions with stationary and moving obstacles. *Psychological review* 102, (4) (1995), pp. 627–651.

[Decugis98] V. Decugis, J. Ferber. *Action selection in an autonomous agent with a hierarchical distributed reactive planning architecture*. In Autonomous Agents'98, pp. 354-361, Minneapolis, USA, 1998. ACM.

[Donikian04] S. Donikian. *Modélisation, contrôle et animation d'agents virtuels autonomes évoluant dans des environnements informés et structurés* (Modelling, control and animation of autonomous virtual agents evolving in structured and informed environments). Habilitation to supervise research in computer science, University of Rennes 1, 26 August 2004, 236 pages.

[Donikian01] S. Donikian. *HPTS: a behaviour modelling language for autonomous agents*. In Autonomous Agents'01, pp. 401-408, Montreal, Canada, May 2001. ACM Press.

[Donikian99] S. Donikian, G. Moreau, G. Thomas. *Multimodal driving simulation in realistic urban environments*. In Progress in System and Robot Analysis and Control Design, pp. 321-332. *Lecture Notes in Computer Science* Vol. 243, Springer, Berlin, 1999.

[Donikian97] S. Donikian. VUEMS: a virtual urban environment modeling system. In *Computer Graphics International'97*, pp 84–92, Hasselt-Diepenbeek, Belgium, June 1997. IEEE Computer Society Press.

[Donikian95] S. Donikian, E. Rutten. *Reactivity, Concurrency, Data-flow and Hierarchical Preemption for Behavioural Animation*. In Eurographics Workshop on Programming Paradigms in Graphics, pp. 137-153, 1995.

[Duckam03] M. Duckham and L. Kulik. .Simplest Paths.: Automated route selection for navigation. In W. Kuhn, M. F. Worboys, and S. Timpf, editors, *Spatial Information Theory: Foundations of Geographic Information Science*, *Lecture Notes in Computer Science* Vol. 2825, pp. 182-199, Springer, Berlin, 2003.

[EAgames99] Maxis Entertainment. The sims. http://thesims.ea.com/, 1999.

[EAgames04] Maxis Entertainment. The sims 2. http://thesims2.ea.com/, 2004.

[Engle99] R.W. Engle, N. Oransky. *The Nature of Cognition*, chapter Multi-Store versus Dynamic Models of Temporary Storage in Memory, pp. 515–555, The MIT Press, 1999.

[Farenc99] N. Farenc, R. Boulic, D. Thalmann. An informed environment dedicated to the simulation of virtual humans in urban context. In *Computer Graphics Forum* (1999), Vol. 18(3), pp. 309–318.

[Ferguson92] I.A. Ferguson: TouringMachines: Autonomous Agents with Attitudes, Computer Laboratory, University of Cambridge, UK, 1992.

[Fernandez02] J.A. Fernandez-Madrigal, J. Gonzalez. Multihierarchical graph search. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):103.113, 2002.

[Feurtey00] F. Feurtey. *Simulating the collision avoidance behavior of pedestrians*. Master thesis report, The University of Tokyo, School of Engineering, 2000

[Fischer96] K. Fischer, J.P. Müller, M. Pischel: A Pragmatic BDI Architecture, in Wooldridge M.J., et al.(eds.), Intelligent Agents II: Agent Theories, Architectures, and Languages, Springer, Berlin/Heidelberg/New York/Tokyo, pp.203-218, 1996.

[Fum07] D. Fum, F.D. Missier, A. Stocco. The cognitive modeling of human behavior: Why a model is (sometimes) better than 10,000 words. *Cognitive Systems Research*, 8(3), 135-142, 2007

[Funge98] J.D. Funge. *Making them behave: cognitive models for computer animation.* PhD thesis, University of Toronto, Canada, 1998.

[Geraerts08] R. Geraerts, A. Kamphuis, I. Karamouzas, M. Overmars. Using the Corridor Map Method for Path Planning for a Large Number of Characters. In Motion in Games MIG2008, *Lecture Notes in Computer Science* Vol. 5277, Springer, 2008.

[Gérin05] M. Gérin-Lajoie, C. Richards, B. McFadyen. The negociation of stationary and moving obstructions during walking : anticipatory locomotor adaptations and preservation of personal space. *Motor Control* 9 (2005), 242–269.

[Gibson79] J. Gibson. *The ecological approach to visual perception*. NJ: Lawrence Erlbaum Associates, Inc, Hillsdale, 1979.

[Gillund84] S. Gillund, R.M. Shiffrin. A retrieval model for both recognition and recall, in *Psychological review*, Volume 91, number 1, 1984.

[Goffman71] E. Goffman. *Relations in public : microstudies of the public order*. New York : Basic books, 1971.

[Goldberg97] A. Goldberg A. IMPROV: A System for Real-Time Animation Interactive Synthetic Actors. In R. Trappl, P. Petta, editors, *Creating Personalities for Synthetic Actors*, *Lecture Notes in Artificial Intelligence* Vol. 1195, 58-73, Springer, Berlin 1997.

[Goldenstein01] S. Goldenstein, M. Karavelas, D. Metaxas, L. Guibas, E. Aaron, A. Goswami. Scalable nonlinear dynamical systems for agent steering and crowd simulation. *Computers and Graphics 25*, 6 (2001).

[Golledge95] R. Golledge. Path selection and route preference in human navigation: A progress report. In A. Frank and W. Kuhn, editors, *Spatial Information Theory: A Theoetical Basis for GIS*, *Lecture Notes in Computer Science* Vol. 988, pp. 207-222, Berlin: Springer, 1995.

[Halperin95] J.R.P. Halperin. Cognition and Emotion in Animals and Machines. In: H. Roitblat, J.A. Meyer, editors: *Comparative Approaches to Cognitive Science*. Chapter 19. MIT Press, Cambrigde MA, 1995.

[Harnad90] S. Harnad. *The symbol grounding problem*. Physica D, 42:335-346, 1990.

[Hartholt08] A. Hartholt, T. Russ, D. Traum, E. Hovy, S. Robinson. *A common ground for virtual humans: using an ontology in a natural language oriented virtual human architecture*. In Proceedings of the Sixth International Language Resources and Evaluation (LREC'08), pp. 3197-3204, 2008.

[Helbing95] D. Helbing, P. Molnar. Social force model for pedestrian dynamics. Physical Review E 51 (1995), 4282.

[Helbing00] D. Helbing, I. Farkas, T. Vicsek. Simulating dynamical features of escape panic. *Nature*, Vol. 407, 2000.

[Helbing05] D. Helbing, L. Buzna, A. Johansson, T. Werner. *Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions*. Transportation Science, 39(1):1–24, 2005.

[Herrero03] P. Herrero, A. de Antonio. *Introducing human-like hearing perception in intelligent virtual agents*. In AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems, pp. 733-740, ACM Press. New York, NY, USA, 2003.

[Hillier93] B. Hillier, A. Penn, J. Hanson, T. Grajewski, J. Xu. Natural movement: or, configuration and attraction in urban pedestrian movement. *Environment and Planning B: Planning and Design 20* (1993), 29–66.

[Hirose02] N. Hirose. An ecological approach to embodiment and cognition. *Cognitive Systems Research*, Vol. 3, N°3, pp. 289–299, 2002.

[Hochmair05] H.H. Hochmair, V. Karlsson. Investigation of preference between the leastangle strategy and the initial segment strategy for route selection in unknown environments. In *Lecture Notes in Computer Science*, 3343:79 . 97, 2005.

[Hostetler et al. 2002] T.R. Hostetler, J. Kearney. Strolling down the avenue with a few close friends. In *Third Irish Workshop on Computer graphics* (2002), pp. 7-14.

[Hudson97] T. Hudson, M. Lin, J. Cohen, S. Gottschalk and D. Manocha. *V-COLLIDE: Accelerated Collision Detection for VRML*. In Proc. of VRML'97.

[Hy04] R. Le Hy, A. Arrigoni, P. Bessière, O. Lebeltel. *Teaching bayesian behaviours to video game characters*. Robotics and Autonomous Systems, 47(2-3):177-185, June 2004.

[Jepson96] W. Jepson, R. Liggett, S. Friedman. Virtual modelling of urban environments. *Presence Teleoperators and Virtual Environments*, 5(1):72-86, 1996.

[Kallmann99] M. Kallmann, D. Thalmann. A behavioral interface to simulate agent-object interactions in real-time. In *Computer Animation'99*, pp. 138-146, Geneva, Switzerland (1999), IEEE Computer Society Press.

[Kallmann01] M. Kallmann, *Object Interaction in Real-Time Virtual Environments*, doctoral dissertation, École Polytechnique Fédérale de Lausanne, 2001.

[Kapadia09] M. Kapadia, S. Singh, W. Hewlett, P. Faloutsos. Egocentric affordance fields in pedestrian steering. In *I3D '09: Proceedings of the 2009 symposium on Interactive 3D graphics and games* (New York, NY, USA, 2009), ACM, pp. 215–223.

[Kavraki96] L. Kavraki, M. Kolountzakis, J.C. Latombe. Analysis of probabilistic roadmaps for path planning. In *Proc. IEEE ICRA* (1996), Vol. 4, pp. 3020–3025.

[Kawasaki99] H. Kawasaki, T. Yatabe, K. Ikeuchi, M. Sakauchi. *Construction of a 3d city map from real-world video and digital maps.* In R. Shibasaki and Z. Shi, editors, International Workshop on Urban 3D/Multi-Media Mapping, pp. 165-168, The University of Tokyo, Japan, September 1999.

[Keijzer02] F. Keijzer. Representation in dynamical and embodied cognition. *Cognitive Systems Research*, Vol. 3, n°3, p. 275-288, 2002.

[Khoo02] A. Khoo, G. Dunham, N. Trienens, S. Sood. *Efficient, realistic NPC control systems using behavior-based techniques.* In AAAI 2002 Spring Symposium Series: Artificial Intelligence and Interactive Entertainment, 2002.

[Kihlstrom99] J.F. Kihlstrom. *The Nature of Cognition*, chapter Conscious versus Unconscious Cognition, pp. 173–203, The MIT Press, 1999.

[Kim05] Y. Kim, M. van Velsen, R.W. Hill Jr. *Modeling dynamic perceptual attention in complex virtual environments.* Lecture Notes in Computer Science, pp. 266-277, 2005.

[Kim05b] Y. Kim, R.W. Hill Jr., D.R. Traum. *A computational model of dynamic perceptual attention for virtual humans.* In AAMAS'05: proceedings of the fourth joint conference on autonomous agents and multiagent systems, pp. 1097-1098, New York, NY, USA, 2005. ACM Press.

[Kintsch98] W. Kintsch. *Comprehension: A paradigm for cognition*. Cambridge University Press, New York, 1998.

[Kuffner 1998] J.J. Kuffner. Goal-directed navigation for animated characters using real-time path planning and control. *Lecture Notes in Computer Science 1537* (1998), 171–179.

[Kuhl84] J. Kuhl. *Volitional aspects of achievement motivation and learned helplessness: towards a comprehensive theory of action control*. In B. A. Maher, editor, Progress in experimental personality research, number 13, pp. 99-171. Academic Press, New York ,USA, 1984.

[Laird03] J.E. Laird, C.B. Congdom. *The Soar User's Manual Version 8.5*. University of Michigan, November 2003.

[Laird08] J.E. Laird. *Extending the Soar Cognitive Architecture*, In Proc. Of the First Conference on Artificial General Intelligence (AGI-08), 2008.

[Laird09] J.E. Laird, C.B. Congdom. *The Soar User's Manual Version 9.1*. University of Michigan, June 2009.

[Lakoba05] T.I. Lakoba, D.J. Kaup, N.M. Finkelstein. *Modifications of the helbing-molnár-farkas-vicsek social force model for pedestrian evolution*. Simulation, 81(5) :339–352, 2005.

[Lamarche02] F. Lamarche, S. Donikian. *Automatic orchestration of behaviours through the management of resources and priority levels*. In Autonomous Agents and Multi Agent Systems, Bologna, Italy, 2002. ACM.

[Lamarche04] F. Lamarche, S. Donikian. Crowd of virtual humans: a new approach for real time navigation in complex and structured environments. *In : Eurographics'04*, edited. by Cani (M.P.) and Slater (M.). – Grenoble, France, september 2004.

[Landauer97] T.K. Landauer, S.T. Dumais. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review* 104:22, 211-240, American Psychological Association, 1997

[Latombe 1991] J.C. Latombe. *Robot Motion Planning*. Boston: Kluwer Academic Publishers, Boston, 1991.

[Lee76] D. Lee. A theory of visual control of braking based on information about time-to-collision. *Perception 5*, (4) (1976), pp. 437–459.

[Lee92] J.R.E. Lee, R. Watson. *Regards et habitudes des passants*, vol. 57-58 of *Les annales de la recherche urbaine*. 1992.

[Lee07] K.H. Lee, M.G. Choi, Q. Hong, J. Lee. *Group behavior from video: a data-driven approach to crowd simulation*. In SCA '07: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, 109--118.

[Lerner07] A. Lerner, Y. Chrysanthou, D. Lischinski. 2007. *Crowds by example*. In Eurographics 2007, Computer Graphics Forum, Vol. 26, N°3, pp 655--664.

[Lord94] R. G. Lord, P. E. Levy. *Moving from cognition to action: A control theory perspective*. Applied Psychology : an international review, 43 (3):335-398, 1994.

[Loscos03] C. Loscos, D. Marchal, A. Meyer. Intuitive behavior in dense urban environments using local laws. In *Theory and Pratice of Computer Graphics (TPCG'03)* (Birmingham, UK, Juin 2003).

[McCarthy69] J. McCarthy, P. J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, 4:463-502, 1969.

[Maes90] P. Maes. *Situated agents can have goals*. In Patti Maes, editor, Designing Autonomous Agents, pp. 49-70. MIT Press, 1990.

[Magnenat-Thalmann90] N. Magnenat Thalmann, D. Thalmann, O. Renault. *A vision-based approach to behavioral animation*. Journal of Visualization and Computer Animation, 1(1):18-21, August 1990.

[Mallot99] H.A. Mallot. *Spatial cognition: Behavioral competences, neural mechanisms, and evolutionary scaling*. Kognitionswissenschaft, Vol. 8, 1999, pp. 40–48.

[Mandler80] G. Mandler. Recognizing: the judgment of prior occurence. *Psychological Review*, Vol. 87, pp. 252–271, 1980.

[Mateas02] M. Mateas, A. Stern. *Architecture, Authorial Idioms and Early Observations of the Interactive Drama Façade*. Technical Report nCMU-CS-02-198, Carnegie Mellon University, december 2002.

[Mateas02b] M. Mateas, A. Stern. A behavior language for story-based believable agents. In *Working notes of Artificial Intelligence and Interactive Entertainment*, AAAI Press, 2002.

[Moreau98] G. Moreau, S. Donikian. *From psychological an real-time interaction requirements to behavioural simulation*. In Computer Animation and Simulation'98, Springer Computer Science, pp. 29-44. Springer-Verlag, 1998.

[Musse99] S. Raupp Musse, M. Kallmann, D. Thalmann. *Level of autonomy for virtual human agents.* In ECAL '99 : Proceedings of the 5th European Conference on Advances in Artificial Life, pp. 345–349, London, UK, 1999. Springer-Verlag.

[Newell90] A. Newell. *Unified theories of cognition*. Harvard University Press, 1990.

[Ngo93] J.T. Ngo, J. Marks. *Spacetime constraints revisited*. In James T. Kajiya, editor, Computer Graphics (SIGGRAPH '93 Proceedings), Vol. 27, pp. 343-350, Anaheim, California, August 1993.

[Noser95] H. Noser, O. Renault, D. Thalmann, N. Magnenat-Thalmann. Navigation for digital actors based on synthetic vision, memory and learning. *Computer and Graphics*, 19:7-19, 1995.

[Noser97] H. Noser, D. Thalmann. *Sensor based synthetic actors in a tennis game simulation*. In Computer Graphics International'97, pp. 189-198, Hasselt, Belgium, June 1997. IEEE Computer Society Press.

[O'Hara00] N. O'Hara. K-d tree neighbour finding for the flocking algorithm. In *Pacific Graphics* (2000).

[Osaragi04] T. Osaragi. Modeling of pedestrian behavior and its applications to spatial evaluation. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 836-843, Washington, USA, 2004. IEEE Computer Society.

[Panne93] M. van de Panne, E. Fiume. *Sensor-actuator networks*. In J. T. Kajiya, editor, Computer Graphics (SIGGRAPH '93 Proceedings), Vol. 27, pp. 335-342, August 1993.

[Paris06] S. Paris, S. Donikian, and N. Bonvalet. Environmental abstraction and path planning techniques for realistic crowd simulation. *Computer Animation and Virtual Worlds*, 17(3-4):325–335, 2006.

[Paris07] S. Paris, J. Pettré, S. Donikian. *Pedestrian reactive navigation for crowd simulation: a predictive approach*. In Eurographics'07, Computer Graphics Forum, 2007.

[Paris08] S. Paris, D. Lefebvre, S. Donikian. *SIMULEM: Introducing Goal Oriented Behaviours in Crowd Simulation*. In PED2008, 4th International Conference on Pedestrian and Evacuation Dynamics, University of Wuppertall, Germany, February 2008.

[Peinado08] F. Peinado, M. Cavazza, D. Pizzi. *Revisiting character-based affective storytelling under a narrative BDI Framework*. In the first joint International Conference on Interactive Digital Storytelling ICIDS 2008, Erfurt, Germany, November 2008, LNCS n° 5334, Springer.

[Pelechano06] N. Pelechano, N. Badler. *Modeling crowd and trained leader behaviour during building evacuation*. IEEE Comput. Graph. Appl., 26(6) :80–86, 2006.

[Pelechano08] N. Pelechano, J.M. Allbeck, and N.I. Badler, *Virtual Crowds: Methods, Simulation, and Control*, Morgan & Claypool, 2008.

[Peschl71] I. Peschl. *Flow capacity of door openings in panic situations*. BAUN, 26(2) :62–67 (1971).

[Peters02] C. Peters, C. O'Sullivan. Synthetic vision and memory for autonomous virtual humans. *Computer Graphics Forum*, 21(4), November 2002.

[Peters03] C. Peters, C. O'Sullivan. *Bottom-up visual attention for virtual human animation*. In CASA '03, 2003.

[Pettré09] J. Pettré, J. Ondrej, A.H. Olivier, A. Crétual, S. Donikian. Experiment-based Modeling, Simulation and Validation of Interactions between VirtualWalkers. In *SCA'09: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, New Orleans, August 2009.

[Pollack92] M.E. Pollack. The Uses of Plans, *Artificial Intelligence*, 57(1), 43-68, 1992.

[Rao92] A. Rao, M. Georgeff. An abstract architecture for rational agents, C. Rich, W. Swartout, B. Nebel, Editors. *Knowledge Representation and Reasoning (KR&R-92)*, p. 439-449, 1992.

[Reich94] B. Reich, H. Ko, W. Becket, N. Badler. Terrain reasoning for human locomotion. In *Computer Animation* (1994), IEEE.

[Relieu98] M. Relieu, L. Quéré. *Les risques urbains: Acteurs, systèmes de prévention*. Anthropos: Collection ville. 1998, ch. Mobilité, perception et sécurité dans les espaces urbains.

[Reynolds87] C. Reynolds. *Flocks, herbs and schools: A distributed behavioral model*. In SIGGRAPH'87, 1987.

[Reynolds99] C. Reynolds. Steering behaviors for autonomous characters. In *Game Developers Conference 1999* (1999).

[Reynolds00] C. Reynolds. Interaction with groups of autonomous characters. In *Game Developers Conference 2000* (2000).

[Rhodes96] B. J. Rhodes. *Phish-nets: Planning heuristically in situated hybrid networks*. Master's thesis, Massachusets Institute of Technology, 1996.

[Russell03] S.J. Russell, P. Norvig. *Artificial Intelligence — A Modern Approach*, Pearson Education, London (2003)

[Schadschneider02] A. Schadschneider, A. Kirchner, K. Nishinari. *CA Approach to Collective Phenomena in Pedestrian Dynamics*. In ACRI 2002, LNCS 2493, pp. 239–248, 2002.

[Scherl03] R.B. Scherl, H.J. Levesque. Knowledge, action, and the frame problem. *Artificial Intelligence*, 144(1-2):1-39, 2003.

[Schmidt00] B. Schmidt. *The Modelling of Human Behaviour*, SCS - The Society for Computer Simulation International, 2000.

[Searle80] J.R. Searle. *Minds, brains, and programs*. Behavioral and Brain Sciences, 3:417-424, 1980.

[Shallice82] T. Shallice. *Specific impairments of planning*. Philosophical Transactions of the Royal Society, 298:199-209, 1982.

[Shao05] W. Shao, D. Terzopoulos. Environmental modeling for autonomous virtual pedestrians. *Digital Human Modeling for Design and Engineering Symposium*, 2005.

[Shao05b] W. Shao, D. Terzopoulos. Autonomous Pedestrians. In *Proc. ACM Siggraph / Eurographics Symposium on Computer Animation* (SCA 05), ACM Press, 2005, pp. 19–28.

[Sierhuis01] M. Sierhuis. *Modeling and Simulating Work Practice. BRAHMS: a multiagent modeling and simulation language for work system analysis and design*. PhD thesis, University of Amsterdam, September 2001.

[Sierhuis03] M. Sierhuis, W.J. Clancey, C. Seah, J.P. Trimble, M.H. Sims. Modeling and Simulation for Mission Operations Work System Design. *Journal of Management Information Systems*, Vol. 19, nÆ4, p. 85-128, 2003.

[Sloman97] A. Sloman. What sort of Control System is able to have a Personality, R. Trappl, P. Petta Editors. Creating Personalities for Synthetic Actors, *Lecture Notes in Artificial Intelligence*, Vol. 1195, Springer-Verlag, p. 166–208, 1997.

[Squire84] L.R. Squire, N.J. Cohen. *The neurobiology of learning and memory*, chapter Human Memory and Amnesia, pp. 3–64. – Guilford, 1984.

[Squire95] L.R. Squire, B.J. Knowlton. *Memory, hippocampus and brain systems*. In The cognitive neurosciences. pp. 825–837, MIT Press, 1995.

[Staller00] A. Staller, S.A. Sloman, T. Ben-Zeev. Perspective effects in nondeontic versions of the Wason selection task. Memory & Cognition, 28(3). pp. 396-405, 2001.

[Sud07] A. Sud, R. Gayle, E. Andersen, S. Guy, M. Lin, D. Manocha. Real-time Navigation of Independent Agents Using Adaptive Roadmaps. In *ACM Symposium on Virtual Reality Software and Technology* 2007.

[Sun03] R. Sun. *A Tutorial on CLARION 5.0.* Technical Report, Cognitive Science Department, Rensselaer Polytechnic Institute, 2003.

[Sun05] R. Sun, P. Slusarz, C. Terry. The interaction of the explicit and the implicit in skill learning: A dual-process approach. *Psychological Review*, 112(1):159-192, 2005.

[Sung04] M. Sung, M. Gleicher, S. Chenney. *Scalable behaviors for crowd simulation.* In Eurographics'04, Computer Graphics Forum, Vol. 23, N° 3, 2004.

[Taatgen06] N.A. Taatgen, C. Lebiere, J.R. Anderson. *Modeling Paradigms in ACT-R*. In Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation. Cambridge University Press, pp 29-52, 2006.

[Taatgen08] N.A. Taatgen, J.R. Anderson. ACT-R. In R. Sun (ed.), *Constraints in Cognitive Architectures*. Cambridge University Press, pp 170-185, 2008.

[Thomas00] G. Thomas, S. Donikian. *Virtual humans animation in informed urban environments*. In Proceedings of the Conference on Computer Animation (CA'00), pp. 112-121. IEEE, may 2000.

[Thomas00b] G. Thomas, S. Donikian. Modelling virtual cities dedicated to behavioural animation", in M. Gross and F. R. A. Hopgood (eds*.), Proceedings of EUROGRAPHICS' 2000,* Vol. 19, (2000), Blackwell Publishers.

[Thomas03] R. Thomas, S. Donikian. A model of hierarchical cognitive map and human-like memory designed for reactive and planned navigation. In *Proc. of the 4th International Space Syntax Symposium* (2003).

[Thomas05] R. Thomas. *Modèle de mémoire et de carte cognitive spatiales: application à la navigation du piéton en environnement urbain*. PhD thesis, Université de Rennes 1, 2005.

[Thomas08] R. Thomas, S. Donikian. A spatial cognitive map and a human−like memory model dedicated to pedestrian navigation in virtual urban environments. Spatial Cognition V: Reasoning, Action, Interaction, *Lecture Notes in Computer Science*, Vol. 4387, pp 421-438, 2008, Springer Verlag

[Thrun96] S. Thrun, A. Bücken. Integrating gridbased and topological maps for mobile robot navigation. In *Proc. of the AAAI Thirteenth National Conference on Artificial Intelligence* (1996), AAAI Press / MIT Press.

[Travers88] M. Travers. Animal construction kits. In artificial life, pp. 421-442. Addison-Wesley, 1988.

[Trésilian91] J. Trésilian. Empirical and theoretical issues in the perception of time to contact. *Journal of Experimental Psychology Human Perception and Performance* 17 (3) (1991), 865–876.

[Tu94] X. Tu, D. Terzopoulos. *Artificial fishes: Physics, locomotion, perception, behavior*. In Computer Graphics (SIGGRAPH'94 Proceedings), pp. 43-50, Orlando, Florida, July 1994.

[Tulving83] E. Tulving. *Elements of Episodic Memory*. Clarendon Oxford, 1983.

[Tulving90] E. Tulving, D.L. Schacter. Priming and human memory systems. *Science*, Vol. 247, 1990, pp. 301–306.

[Turner02] A. Turner, A. Penn. Encoding natural movement as an agent-based system: an investigation into human pedestrian behaviour in the built environment. *Environment and planning B: Planning and Design 29* (2002).

[Uexküll56] J. von Uexküll. *Streifzüge duch die Umwelten von Tieren und Menschen*. Hamburg, Germany, Rowohlt, 1956.

[Ulicny02] B. Ulicny, D. Thalmann. *Towards interactive real-time crowd behavior simulation*. Computer Graphics Forum 21, 4 2002, 767–775.

[Urban01] C. Urban, B. Schmidt. PECS - Agent-Based Modelling of Human Behaviour. *Emotional and Intelligent II - The Tangled Knot of Social Cognition, AAAI Fall Symposium,* AAAI Press, 2001.

[Vernon07] Vernon D., Metta G., Sandini G.: A Survey of Artificial Cognitive Systems - Implications For The Autonomous Development of Mental Capabilities in Computational Agents, *IEEE Transactions on Evolutionary Computation*, 11(2):151-180, 2007.

[Warren95] W.H. Warren. Perception of Space and Motion, chapter *Self-Motion: Visual Perception and Visual Control*, pp. 263–325, Academic Press, 1995.

[Wilhelms89] J. Wilhelms, R. Skinner. *An interactive approach to behavioral control*. pp. 1-8, June 1989.

[Wooldridge00] M. Wooldridge. *Reasoning about Rational Agents*, MIT Press, 2000.

[Yu07] Q. Yu, D. Terzopoulos. *A decision network framework for the behavioural animation of virtual humans*. In D. Metaxas and J. Popovic, editors, Eurographics/ACM SIGGRAPH Symposium on Computer Animation, pp. 119-128, 2007.

[Zhao99] H. Zhao, R. Shibasaki. *A system for reconstructing urban 3d objects using ground-based range and ccd sensors.* In R. Shibasaki and Z. Shi, editors, International Workshop on Urban 3D/Multi-Media Mapping, Tokyo, Japan, September 1999.

[Ziemke02] T. Ziemke. Introduction to the special issue on situated and embodied cognition. *Cognitive Systems Research*, Vol. 3, n°3, p. 271-274, 2002.