INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

MEEMOs: Believable Agents with Episodic Memory Retrieval

Paulo Manuel Fontaínha Gomes

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática e de Computadores

Júri

Presidente:	Doutor José Manuel Nunes Salvador Tribolet
Orientador:	Doutora Ana Maria Severino de Almeida e Paiva
Co-Orientador:	Doutor Carlos António Roque Martinho
Vogal:	Doutora Maria da Graça de Figueiredo Rodrigues Gaspar

Setembro 2010

Acknowledgements

I would like to thank Ana Paiva and Carlos Martinho for their support, guidance, and countless corrections on the thesis. I express my gratitude to João Dias for pointing out several important memory related research sources and for allowing me to use FAtiMA's source code. I thank Eurico Dourado for the extensive feedback concerning the thesis evaluation, the suggestions during this document's writing process, and for the time and effort put into brainstorming ideas for the dissertation. I express my gratitude to Juliana Pantaleão, Andreia Catarrinho and Pedro Branco for allowing me to use and extend the application that supported the evaluation. I would like to thank Joana Campos for her companionship and for pointing out very useful memory related research sources. I express my gratitude to Marco Vala for his commentaries and suggestions concerning the evaluation's report. I thank my close friends for enduring my work related grumpiness. Finally, I would like to thank my parents and my brother for the uplifting remarks that got me by writing this thesis.

Porto Salvo, September 28th 2010
Paulo Manuel Fontaínha Gomes

to my parents

Resumo

O objectivo desta tese era desenhar um modelo para agentes que recordassem episódios emocionalmente importantes. Queríamos também aferir se incluir esse modelo numa arquitectura melhoraria a credibilidade dos seus agentes. Considerámos alguns conceitos base de memória humana, avaliação emocional e credibilidade. Em seguida analisámos arquitecturas de agentes que suportam memórias e emoções, apercebendo-nos que nenhuma preenchia todos os nossos requisitos para recuperação de memória. Prosseguimos com a descrição do nosso modelo dividido em dois processos: *despontar*, em que os estímulos percepcionados (pistas de recuperação) são correlacionados com as memórias; e *vivência da lembrança*, em que as memórias para as quais houve uma forte correlação são avaliadas emocionalmente. Propusemos uma aproximação de despontar em que os locais servem como pistas de recuperação indirectas (despontar localizado). Descrevemos a implementação do nosso modelo e como a usámos para controlar o comportamento de personagens num vídeo jogo. Implementámos o despontar localizado, e a vivência da lembrança foi criada a partir de um sistema de avaliação emocional ao qual foram adicionadas regras de reacção para eventos gerados pelo despontar. Gravámos vídeos da aplicação e usámo-los para desenhar uma avaliação não interactiva do modelo. A estrutura da mesma era entre grupos, com pré-teste e pós-teste, tendo participado 96 indivíduos. Os resultados são consistentes com a nossa hipótese de que agentes modelados na nossa arquitectura seriam vistos como mais credíveis do que agentes modelados em arquitecturas semelhantes mas sem recuperação de memórias episódicas.

Abstract

The objective of this thesis was to create a model for agent memory retrieval of emotionally relevant episodes. Additionally we wished to assess if including such a system in an architecture would improve the agents' believability. We reviewed some core concepts concerning human memory, appraisal theories and believability. Then we analyzed agent architectures that support memory and emotions, realizing that none fulfilled all our requirements for memory retrieval. We proceeded by describing our retrieval model consisting of two main steps: *ecphory*, in which the perceived stimuli (retrieval cues) are matched with memories; and *recollective experience* that re-appraises memories that had a positive match. We proposed a location ecphory approximation, in which locations serve as indirect retrieval cues. We described how we implemented our model and used it to drive the behavior of characters in a game application. We implemented the location ecphory approximation, and the recollective experience consisting of a reactive appraisal module with reaction rules for events generated by the former. We recorded the application running and used the videos to create a non-interactive evaluation. The evaluation's structure was a between-groups pre-test/post-test one with two control groups, and we had 96 participants. The evaluation's results are consistent with our hypothesis that agents modeled by our architecture would be perceived as more believable than agents modeled in similar architectures without episodic retrieval.

Palavras Chave
Keywords

Palavras Chave

Recuperação de Memória
Memória Episódica
Credibilidade
Emoções
Avaliação Emocional
Despontar

Keywords

Memory Retrieval
Episodic Memory
Believability
Emotions
Appraisal
Ecphory

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

What makes a synthetic agent believable? What are the features, the attributes, an agent should have in order to increase its believability? Scientists in the field of synthetic autonomous agents have been trying to answer these two questions for decades. In this dissertation we will try to address these questions by identifying one of these features: retrieval of emotionally relevant episodic memories. Episodic memory has been defined as "a type of long-term memory for personal experiences and events ... occurring at particular times and places" [12].

To demonstrate the importance of these memories' retrieval, a common episode of everyday life will be explored: forgetting where one has left one's keys. Take the following hypothetical example of Michael: just before going to work in the morning, he is unable to remember where he put his keys the day before. He looks in the hall table, the place where he usually drops them, yet the keys are not there. He actually left them in the kitchen on top of the fridge. The problem is that he cannot accurately recall an episode of his life, with an associated time and place: the episode is "leaving his keys"; the time is "the day before"; and the place is "on top of the fridge in the kitchen". Now imagine that Michael could never remember were he left his keys, or any other door-opening devices. His everyday life could become very frustrating (and the task of opening doors a lot harder).

Taking the example even further, imagine that Michael could not recall a single episode of his life. He would still know his name, would still know that on a clear sunny day the sky is generally blue and how to eat with a fork and a knife. Nonetheless, he would not be able to recall what had happened to him the day before, or any day for that matter. Not being able to remember his life's episodes would unavoidably influence his behavior. Michael would not take notice of the place where he had his first kiss as he walked by it. He would not understand most of the jokes his friends would make about trips he did or events he witnessed. The reason for this happening would be Michael's inability to recall life episodes, and their corresponding time and place. In particular, episodes that were emotionally relevant to him. He no longer reacted emotionally to references of important events of their shared lives. The scenario is fictional, and some might even recall the film Memento (Nolan, 2000) where the main character is unable to retain long-term memories. But reality is closer to fiction than might be expected. Among other clinical cases of impaired episodic memory [30][55], the case of K.C. [53] has some

resemblances with the just presented scenario. At the age of 30, K.C. suffered head injury due to a motorcycle accident. His episodic memory was almost completely lost, yet his remaining cognitive capabilities remained untarnished.

Having explored our hypothetical scenario, it is easy to agree with the idea that the absence of episodic memory can severely affect a person's life. One might ask if this idea can be transposed to the area of synthetic agents. A question that can be set forth is the following: "If synthetic agents were to have episodic memory, would that improve their believability?". This question is particularly relevant as many contemporary synthetic agents are still memory impaired, at least in what concerns episodic memory.

A new scenario can help in the exploration of this possibility. Consider a role-playing video game such as Dragon Age[1] or Final Fantasy XIII[2], in which the player controls an avatar that is accompanied by a small group of computer controlled characters (ally characters). We will focus on these ally characters as synthetic agents.

At critical junctures in the game, the player faces boss enemies. These bosses demand a reasonable effort to be vanquished, and in general are important elements in the game story. Moreover, in general either the avatar, or the non-player characters of the group, display some sort of emotional reaction when the mighty foe is defeated: relief for no longer being in danger; pride for defeating the difficult adversary; among others.

Later in the game, after having defeated the boss, the player can revisit the location where the battle took place. This may happen because the player is exploring the game world or because the location is in the path between new game objectives. When passing in such a location, the avatar and ally characters do not express any emotional reaction concerning what has happened there during game-play. Although the location is significant in the game story, the characters behave as it was simply an "empty room". The player remembers, but the characters do not.

On one hand, characters react emotionally to the event of defeating the boss. On the other, their behavior does not seem to change with experience. This lack of reaction may be seen as incoherent with the initial emotional reaction. The problem is that characters do not remember the episode of defeating the boss. If they were to remember this episode, many possible reactions could be expressed: bragging about vanquishing the enemy; happiness as the victory pushed the group closer to a greater goal; among others.

The problem can be generalized to other important episodes that elicit reactions from the characters: a new character joining the group, a character leaving the group, a new item being acquired, and so on. The characters do not seem to remember events that occurred during game-play when passing by the locations where these events took place. Characters sometimes display reactions towards a game world location, but it is evident that this is scripted behavior, rather than the retrieval of a personal experience.

This problem is also present in games with a "sandbox" level design approach, such as Grand Theft Auto IV[3]. As the player roams through a self contained open world, he or she is bound to go by locations were relevant events have previously occurred. Should not the avatar display

[1] by Bioware ©EA International (Studio and Publishing) Ltd.
[2] ©SQUARE ENIX CO.,LTD
[3] ©Rockstar Games

a behavior that transmits it is aware of the location's personal relevance? All these examples illustrate the importance of creating an architecture for believable agents that supports retrieval of emotionally relevant episodic memories.

1.2 Problem

In this dissertation we will analyze the possibility of integrating episodic memory into an agent architecture. More specifically, we will tackle the following problem:

> How can one create autonomous agents that remember episodic memories that were emotionally relevant to them?

The approach described in this thesis will be to model episodic memory retrieval as a two stage process: in the first stage, the agent's current perceived stimulus are matched against its stored memories; in the second stage, the memories that were selected on the first stage, based on the matching value, are relived by the agent. The agent relives the memories by re-appraising emotionally the past events these memories are linked with. Consequently, we analyzed agent architectures that model appraisal processes as well agent architectures that focus on memory.

As our approach was inspired on human memory research, we will present the main psychological concepts that guided our work. However, our goal is not to mimic natural processes. We are mainly concerned with the believability of agents, not with their realism. Therefore, in the architecture some processes that are not the thesis' focus will be a simplification of memory theory, while others will simply not be modeled. In light of this, our main hypothesis is the following:

> Autonomous agents modeled by an architecture that incorporates episodic memory retrieval of emotionally relevant events, will be perceived as more believable, than agents modeled by a similar architecture that does not incorporate episodic memory retrieval.

Finally, it should be noted that the tackled problem does not concern learning. We are not interested in modelling a utilitarian use of memories nor their use as a survival mechanism. What we are indeed concerned with is agent's emotional reaction to the elicitation of memories.

1.3 Outline

This document is organized in 6 chapters.

Chapter 2 (Background) gives a superficial overview of background theory concerning psychology of human memory, appraisal theories of emotion, and believability of synthetic characters. Concepts such as episodic memory, memory retrieval, appraisal, and believability, are explored.

Chapter 3 (Related Work) we analyze architectures for autonomous agents with long-term memory, and architectures for autonomous agents with emotions. The relevance of these architectures to our current work will be assessed.

Chapter 4 (Model) our model for an architecture supporting episodic memory is presented. Particular focus is given to episodic memory retrieval.

Chapter 5 (Implementation) we describe how we implemented the architecture described in the previous chapter, and the application necessary for the model's evaluation.

Chapter 6 (Evaluation) the methodology used to evaluate the model is presented and the results from gathered data are analyzed.

Chapter 7 (Conclusion) we present some conclusions that can be drawn from the analyses of the data concerning this dissertation's problem. Last, we describe possible paths for future work.

Chapter 2

Background

In this chapter we present some of the background theories, from different areas, that motivated, and supported, many of the research decisions taken in this dissertation. Firstly, we review some core concepts of psychology and neurology concerning human memory, focusing on episodic memory. Afterwards we give a short introduction to appraisal theories in general, followed by an overview of one of these theories. Then, we describe how the concept of believability has been studied by artists and computer science researches. To conclude, we point out how the referenced background theories influenced this dissertation.

2.1 Memory

Human memory is the mental process by which we encode, store and retrieve information [54]. Although definitions may vary, in general the three processes referenced (encoding, storing and retrieving) are usually presented as being part of the memory mechanism. The encoding process consists of the perception of stimulus, creation of a memory trace ("a physical representation of a memory in the brain" [12]), and initial registration of this trace. The storing process refers to the retention and maintenance of information in memory over time. Retrieval consists in the recovery and use of the stored information.

The idea that certain memories are only kept for a short period of time, and others last much longer, is transversal to almost all memory theories. The Atkinson & Shiffrin Model [29] was seminal in the area of human memory and materializes this idea. According to the model, memory can be divided into three entities: Sensory Register (SR), Short-term Memory (STM) and Long-term Memory (LTM). These three elements are considered to be part of a pipeline for encoding and storing information. First, the sensory register receives sensorial data and keeps it for a very short-period of time (less than a second to around three seconds). Second, information is transferred to STM by pattern recognition or attention focus. STM has limited storage space: humans tend to be able to store around seven items of information in it [39]. Information items are kept in STM for a few seconds, but this duration may vary due to rehearsal. Rehearsal can be done by mental or verbal repetition of the items. Third, items in STM get to LTM by elaborative rehearsal, that is, new items are linked with previously existing information. In this process old information can be changed and recoded in a different way. LTM stores large amounts of data over long periods of time and we will examine it in greater detail.

It is generally acknowledged among researchers in the field of psychology and neuroscience that LTM is made of different systems interacting together. If these systems actually correspond to different subsystems of the brain is still an open question [53]. LTM can be divided in three systems: procedural memory, semantic memory and episodic memory. Procedural memory refers to rudimentary skills, cognitive or physical, that humans tend to acquire by practice (e.g. how to eat with a fork and a knife). Semantic memory refers to general knowledge about the world and facts one knows (e.g. on a clear sunny day the sky is generally blue). Episodic memory refers to personal experiences that are linked with a specific time and place (e.g. I left my keys on top of the fridge in the kitchen yesterday night). Episodic and semantic memories are often analyzed together as autobiographic memory [3]. Barclay claimed that autobiographic memories can change with time. The reason proposed for this change was that repeatedly experiencing similar events may lead to generalization: after a person has experienced similar events these might be combined in a generalized schema. A person might not be able to recall individual events, but rather be able to reconstruct how an event type usually takes place. In this way, episodic memories are combined together, and generalized to semantic knowledge.

Focusing on episodic memories, Tulving [53] stated that these memories enable humans to do mental time travel, that is, to relive past experiences. A person can relive a past event as an observer or as an actor [38]. The re-experience takes place during retrieval, that has been defined as a process in which memory traces interact with retrieval cues, stimulus that can be either internal or external [49]. In [54], retrieval is described as a two staged process. First, available cues are correlated with memory traces, which might involve search among these traces. This first stage is called ecphory. The product of ecphory is a set of pairs of highly correlated cues and traces. These pairs, ecphoric information, are then selected according to the type of retrieval, and converted into a recollective experience. This second stage is called conversion, and it is through the recollective experience that a person is able to relive a past event, although not as intensely as before [38].

Information kept in episodic memory may not stay there indefinitely. The theories of forgetting deal with this fact in terms of generic LTM. Two of the concepts present in many of these theories are decay and interference. By decay we mean that memories fade with time. Interference accounts for the conflict between old and new information stored. Interference is generally seen through two perspectives: old memories inhibit the acquisition of new ones (proactive interference); new memories cause problems with the recollection of old ones (retroactive interference). Many theories of forgetting have been developed up to date but none of them currently gathers enough support, backed by experimental results, to be considered the prevailing one [56]. In the case of episodic memories, there seem to be at least two contributing factors for a memory to last: rehearsal, meaning that memories not accessed may have a tendency to be forgotten; and emotion arousal caused by the original event, as emotionally more intense events will be harder to forget [38].

2.2 Appraisal Theories

Appraisal theories essentially claim that emotions are elicited by the evaluation of events and situations [48]. This evaluation (appraisal) is done in regard to the circumstance's significance

to the individual [17]. Moreover, this significance to the individual is classified according to dimensions such as: pleasantness, certainty, novelty, agency, coping potential, compatibility with standards, among others. For example, an individual appraises the situation of lying in the sun at the beach as highly pleasant. Suddenly a dog runs over him chasing a ball. This later event is perceived by him as novel and unpleasant. He can further appraise the situation as being inconsistent with the standard of "not perturbing people relaxing" and additionally regard the owner of the dog as the fundamental agency element in the situation.

As we can see in the described scenario, a circumstance's evaluation is the result of the combination of individual appraisals [17]. The creation of these appraisals is generally believed to be a continuous and automatic process. Patterns of appraisals can be correlated with emotions [48]. Returning to our example, after being rundown by the dog, the individual feels a reasonable degree anger and even yells at the dog owner. In this case, the appraisal pattern of perceiving an event as unpleasant, inconsistent with a standard and having agency in another individual, can be correlated with the anger emotion towards this other individual.

Appraisals are usually viewed as being correlated not only to emotions, but also to an overall valence of the emotional state: a mood. An individual's mood may be positive or negative, even in the absence of active defined emotions [17]. Continuing our example, the individual may stay in a negative mood long after the anger feeling has ceased. Equally important, is the idea that emotions can serve as an adaptation mechanism, generating action tendencies, but not being strictly attached to any specific behavior [19]. Emotions are seen as influencing the choice of certain behaviors over others, rather than strictly defining reactions. Finally, it should be noted that the just presented features are common in appraisal theories, still, individual theories may diverge in certain aspects.

In the context of Affective Computing, one well known theory of appraisal is the OCC model [43]. This model defines emotions as a reactive evaluation, conscious or not, of perceived events, objects or agents. Emotions are grouped into emotion types. Each emotion type represents very similar emotions, or emotions states, that differ mainly on their intensity (e.g. emotion type: fear; emotional states: concern, fright, petrified ...). One of the main tenets of the model, is that a situation's evaluation focus determines the emotion type of felt emotions. This focus can be on one of three main aspects of a situation: consequences of an event, causing event-based emotions; agency element of the situation, causing attribution emotions; or an object, causing attraction emotions. Take the following example: a father witnessing his daughter's graduation ceremony. The father, perceiving the event as important for her daughter's career, would be happy-for her (focus on the consequences of an event). He might also have a feeling of admiration towards her, as she spent a great deal of effort to graduate (focus on the daughter as an agency element). Finally, he may enjoy seeing the decorated building serving as background to the ceremony, for its neo-classical style appeals to the father's aesthetic values (focus on the building as an object).

Furthermore, event-based emotions are divided in two groups: those in which the event consequences are for other entities (e.g. pity), and those for which the consequences are for the individual itself (e.g. joy). In turn, emotions in the last group are differentiated according to whether the evaluation took the event as a prospect (e.g. fear), or not (e.g. distress). Finally, some emotions are said to result from evaluating a situation focusing on consequences and agency

at the same time (e.g. anger). Figure 2.1 presents the just described structure.

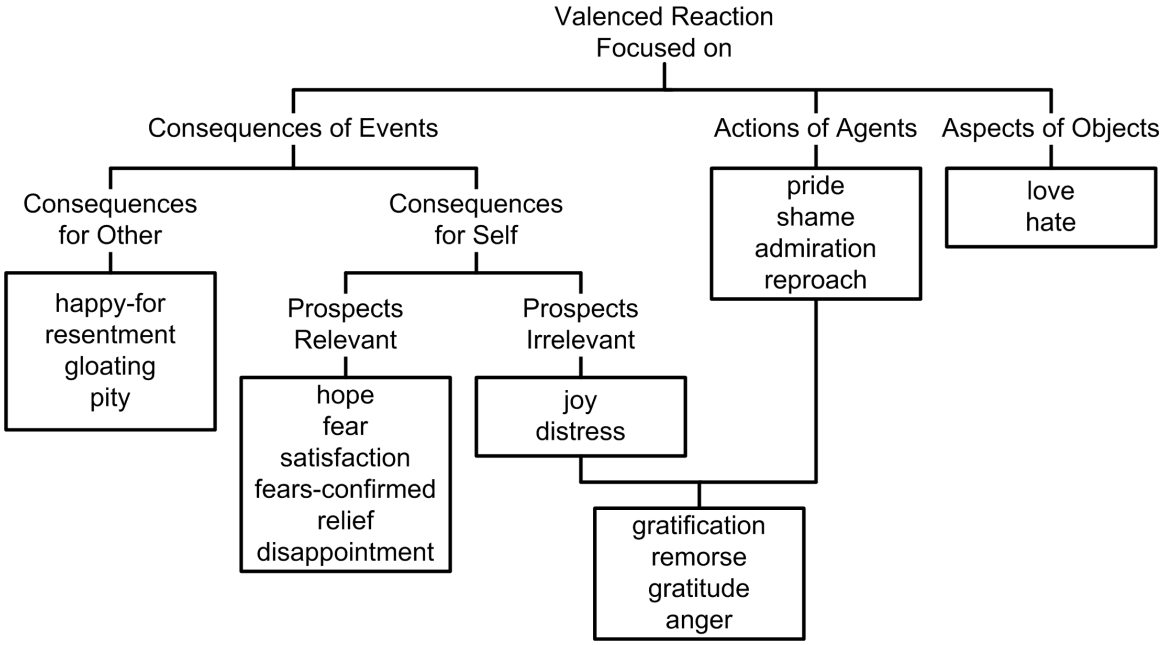


Figure 2.1: Structure of OCC emotion types (simplified from image in [43])

The authors present several factors that can influence emotions' intensities. Most of them can be interpreted as appraisal dimensions. A crucial factor in the model is desirability. Desirability of a situation is evaluated in regard to goals and is only relevant for event-based emotions. According to the OCC model, in people's minds goals are connected in a lattice of causal dependencies. In certain cases a group of goals must be achieved in order for another one to be attained (sub-goals). This structure of inter-dependent goals is constantly changing: goals are realized; goals are abandoned; new goals are introduced. The time a goal stays in the goal structure, that is, before it is discarded due to realization or another reason, depends on its type. The authors classify goals according to threes types: Active-pursuit goals, which a person actively tries to obtain; Interest goals, which a person does not pursue because he or she has little control over their realization; Replenishment goals which are not discarded after being achieved, and have a cyclical nature. Interest goals tend to have a long longevity in the goal structure. All these goals influence the perceived desirability, or undesirability, of a situation: if it contributes to the fulfillment of a goal, it is perceived as desirable; if it may harm the achievement of a goal, it is perceived as undesirable.

Besides desirability, two other factors are central to the model: praiseworthiness and appealingness. Praiseworthiness is the extent to which a certain situation is in accordance with a person's standard and is only considered for attribution emotions. Appealingness of a situation is evaluated in regard to a person's predispositions (e.g. like neo-classical style buildings), and is only relevant for attraction emotions. Desirability, praiseworthiness and appealingness are said to be local factors, as each one only affects the intensity of a group of emotion types. Global factors, affecting the intensity of all emotion types, are also presented. One of the mentioned global factors is proximity. Proximity is a subjective measure of how close a person feels to the evaluated entity. This perceived psychological distance can be, among other things, spatial.

2.3 Believability

In the beginning of the 19th century, the english poet Samuel Taylor Coleridge coined the term *suspension of disbelief* [10]. The term referred to the mental state in which the reader of a poetic piece could regard a supernatural, or simply romantic, character as real, regardless of characteristics out of the ordinary. When explaining his motivation for *Lyrical Ballads* [11], Coleridge expresses his desire to write in "semblance of truth" and to spur the reader's imagination, clouding what would, at first glance, and without context, be regarded as unrealistic.

However, Coelridge was neither the first, nor the last to pursue suspension of disbelief in art. One can argue that the idea has been present throughout art history. Since Coelridge's *Biographia Literaria*, the term has evolved: the concept of character was generalized to a fictional situation; and the term now encompasses a generic art form, and not specifically poetry. One such art form is Animation, that in the 1930's saw great technical, and aesthetic progress, in the hands of Walt Disney Studio artists. Thomas and Johnston would describe the animation principles learned by these artists in the seminal *The Illusion of Life: Disney Animation* [52]. In this book, they describe how animated characters can give the illusion of being alive, of having motivations, of thinking and acting accordingly. At the time of publication, 3D computer animation was taking its first steps. Later on, Lasseter identified the possibility of applying the lessons of traditional animation to this new area [31].

Towards the end of the 20th century, computer science scientists in the field of autonomous agents, began to analyze how the artistic principles of animated characters could be used to design believable agents [4]. These agents would have the characteristics that gave animated characters "the illusion of life". Bates revisited Thomas and Johnston's work, and together with the rest of the OZ Group, developed the *Edge of Intention*. *Edge of Intention* is a simulated world with three self-animated believable agents, called Woggles, that interact between themselves. Three of the key ideas that guided Bates were: an agent's emotional state must be clearly defined; the agent's actions must express what it is thinking about and its emotional state; emotion expression should be accentuated and carefully timed in order for the viewer to clearly acknowledge and savor it.

The definition of believable agent was further dissected by Loyall, also working in the OZ Group [33]. He proposed several requirements for an agent to be believable, much of them orbiting the idea of personality. By personality he considers "all of the particular details - especially details of behavior, thought and emotion - that together define the individual" (p. 16). The other feature requirements were: agents should display emotions coherently with their personalities (*emotion*); agents should be deliberative, act on their deliberation and be pro-active (*self motivation*); agents should grow, and this growth should be aligned with their personalities (*change*); agents should have social relations between themselves aligned with their personalities (*social relationships*); all possible means of physical expression should be consistent with the agent's thought process (*consistency of expression*); agents should appear to have goals and, or, desires (*appearance of goals*); agents should pursue concurrent goals and have parallel actions; agents should be able to react in a reasonable timing according to their personality (*reactive and responsive*); agents should change their behavior according to the situation they are in (*situated*); agents should have limits to what they can physically and mentally do (*resource*

bounded); agents should act in the context of social conventions and cultural elements (*social context*); agent capabilities should be diverse, ranging from sensing to actuating in dynamic worlds (*broadly capable*); agents' different capabilities should have a similar level of complexity and detail (*integrated capabilities*); Boundaries between different types of behaviors should not be noticeable (*integrated behaviors*).

Agent believability was also analyzed in more specific contexts, such as pedagogical agents [32]. Lester and Stone define believability as the identification, from the user or spectator, of an agent's goals, beliefs and personality. They claim that visual qualities of the character, along with the way behavior is sequenced, are the two main factors affecting believability. They also propose three techniques for believability enhancement: *situated liveliness*, *visual impact* and *complex behavior patterns*. Agents should show they perceive the world around them (situated liveliness). Behaviors with higher visual impact, such as moving from one place to another, drive attention and thus enhance believability. Lastly, complex behavior patterns are important in long term interactions with users, as they become harder to recognize. Behavioral pattern recognition can harm believability and unexpected events can sometimes enhance it. Proof of this last idea was the attention shrimp woogle got when performing an unexpected behavior [4], although it was bug related.

Although in many cases not being strictly believable agents, believable characters in video games share characteristics with the former. Rollings and Adams [47] propose that believable video game characters, specially main characters, should: be interesting and intrigue the player; be likable; grow with the game story and overall game experience (pp. 134–135). It is pointed out the difficulty in achieving this last feature. Regarding the second point, although game designers should be careful with creating unlikable characters in a game, the main character's archenemy may spur negative sentiments by the player [28] (p. 245–246). Finally, it can be argued that in spite of believability being correlated with both interest and likability, these two factors may be caused by believability, rather than being the cause of enhanced believability.

Lastly, Ortony [42] proposed a definition for believable agents centered around emotion. He considers that there should be consistency in the way agents evaluate events, and how this evaluation influences their emotional state. The idea is that for similar situations, the emotional reactions generated, and their intensities, should also be similar. He states that behavioral expression of the emotional state should also be consistent in the same sense. Nonetheless, he states that there should be some variability as well, in line with Lester and Stone's view of predictability. Variability should be enough that agents do not appear to act, and react, always in the same way, but not to much as to render the choices arbitrary. Moreover, a second party should be able to recognize the "character". One can interpret this character as the general rules by which the agent evaluates and behaves, despite of small variabilities. Additionally, evaluation and behavior should be coherent across different types of situations and over the agent's experience. Taking the last point, one can infer that the agent's behavior should in principle reflect what it has lived. Concerning the former point, Ortony proposes modelling personality traits and dimensions as a way to achieve coherence across situations.

2.4 Concluding Remarks

We have defined what we understand by episodic memory and its connection with autobiographic memory. In the following chapter we will analyze how these concepts have been used in synthetic agents' mental models. We also saw how episodic memory retrieval can be interpreted as a two stage process: ecphory in which retrieval cues are correlated with memory traces; and recollective experience in which past are relived. This interpretation of episodic memory retrieval drives our agent model presented in Chapter 4.

The fundamental principles of the theories of appraisal were reviewed, in particular, that emotions are elicited by the evaluation of events and situations. We gave an introduction to the OCC model. This model guides many of the architectures for believable synthetic agents presented in the following chapter, as well as our own model. Two factors that influence emotion's intensities were described with greater detail: desirability and proximity. These concepts will be revisited in Chapter 4.

Finally, we presented several characteristics that positively influence a character's believability. These characteristics not only guided the implementation of our model (Chapter 5), but also later served as benchmarks for agent believability in our tests (Chapter 6).

Chapter 3

Related Work

In the following section we present architectures for autonomous agents with autobiographic memory and two architectures that support episodic memories. Afterwards, in Emotion Architectures, we will analyze architectures for autonomous agents with emotions. Last, we compare all presented architectures focusing on the dissertation's problem.

3.1 Memory Architectures

Although our focus is on episodic memories, analyzing agent architectures that support autobiographic memories may serve as a good starting point for developing our multi-agent architecture. Autobiographic memory can be interpreted as a combination of semantic knowledge and episodic memories, so looking into these architectures may indeed be helpful. Three different approaches to the problem of modelling autobiographic memories will be presented: work by Ho, Dautenhahn and Nehaniv [24] on the area of artificial life; FAtiMA's memory episodes [14][25]; and an agent architecture proposed in [26], grounded on goal discrepancy minimization (Discrepancy Architecture). Subsequently, we analyze the agent episodic memory architecture designed by Brom and Lukavsky [8]. Last, we describe SALT (Schema-Associative Model of Memory) [5][6] that also supports episodic memories.

3.1.1 Artificial Life

Ho, Dautenhahn and Nehaniv have developed several autobiographic memory architectures in the context of Artificial Life [24]. The common scenario in all of them is a world where agents need to maintain homeostatic variables in a certain value range in order to survive. Agents achieve this by using resources in the world. A memory system enables agents to go back to a location where a currently needed resource was found.

Trace-Back and Locality are two of the mentioned architectures. They are analyzed in [23] and they are both reactive. In the Trace-Back architecture the agent stores a finite number of ordered memories. Memories are stored whenever the agent encounters an object (Event-Based) or at fixed time steps (Time-Based). Memory items have position and orientation information and can be seen as path steps. If a resource is needed, and is linked with a memory item, a trace-back process starts: the agent undoes the steps in memory ranging from the resource

related memory to the last inserted one. In the Locality architecture, information on how to get from one type of resource to another is kept.

Ho, Dautenhahn and Nehaniv developed two other more complex agent architectures: Short-term Memory Architecture(STMA) and Long-term Autobiographic Memory Architecture (LTMA). STMA is relatively similar to the Trace-Back architecture with event-based memory items. The main difference lies on the number of memory entries kept. For all items in memory, the system verifies if the cost of undoing all steps that have been performed since that item (in the memory sequence) would lead a homeostatic variable to fall outside the acceptable range. If this condition is verified, the item is removed, as well as all previous items in the memory sequence.

In contrast, LTMA's memory is stored as a sequence of Event Specific Knowledge records. Each record has information about which was the resource sensed, about the situation where it was sensed and how the object can influence homeostatic variables. When a homeostatic variable falls out of the acceptable range, the event reconstruction process begins.

First the system searches records for the ones that refer to resources that improve the problematic variable. Each of these records is called a key record. For each key record the system does a backward trace and a forward trace. The backward trace identifies the sequence of actions that can be redone in order to reach the key record state. The forward trace identifies the sequence of actions that can be undone in order to reach the key record state. These action sequences are then filtered according to key record attributes, and ranked according to the total change in homeostatic variables. The top ranked action sequence is then redone, if it was generated from a process of backward trace, or , if it was generated from a process of forward. This enables the agent to reach the needed resource and reestablish the value of the problematic homeostatic variable.

3.1.2 FAtiMA's Memory

The architecture FAtiMA models long-term autobiographic memories [14][25]. Memories are organized into episodes. Each episode can refer to several actions/events that took place in the same location and in the same time-frame. Each episode corresponds to a significant personal memory of the agent that fits a sequence of memories that constitutes a life story.

Episodes can be divided in three components: Abstract, Narrative and Evaluation. The Abstract describes the type of episode and its impact on the emotional state of the agent. Narrative has details about the event, such as location, time of occurrence, who were the agents involved and the cause-effect actions. Evaluation refers to the interpersonal consequences of the evaluation of the cause-effect actions. Cause-effect actions are the most important elements of episodes. They describe the connection between an event or action and the emotion it arose on the agent.

A new episode is generated when the agent changes location or after a time-out, so that each episode is related with a specific time and place. Every time a new episode is created, it becomes the active episode. There is only one active episode at a given time. When actions and events are perceived, they are stored in the active episode. In what concerns events, Dias, et al. [14] make a distinction between the ones occurring outside the agent (external) and mental events connected with goals and intentions (internal).

Memory episodes, either referent to external or internal events, are used when the agent wants to generate a summary of its past history. The most important events are gathered according to the intensity of the emotions they elicited, and sent in a time ordered sequence to a natural language system. This system is responsible constructing a textual story out of the ordered event information.

3.1.3 Discrepancy Architecture

In [26] the authors describe an architecture for believable agents that supports autobiographic memories. Memory retrieval is presented as a reconstructive process: autobiographic memories are created by the combination of individual episodic memory elements. Agents try to minimize the discrepancy between their goal state and their current state, and memory retrieval supports this mechanism.

Autobiographical knowledge is applied in four different processes: goal formulation, goal attainment verification, event coping, and event verification. The agent's current goals can be extracted from the autobiographic knowledge (goal formulation), so that in order to verify if they were achieved (goal attainment verification) the autobiographic knowledge module is also used. When an event occurs, the system tries to match it against passed experienced events (event verification). If a positive match is found, the strategy used in the past is re-used or adapted (event coping). If not, the agent will resort to emotion-based coping or to a more rudimentary instinctive behavior. The perceived event and the way the agent dealt with it are encoded in a single unit. This memory unit is stored in the autobiographic memory and can be combined with similar units. The combination of memory units generates general knowledge and reduces memory storage.

3.1.4 Full Episodic Memory

In [8] an architecture for agents with full episodic memory is presented. Full in the sense that the agent stores in memory a representation of the majority of elements perceived, filtered according to an estimate of their relevance to the same agent. The concept of full episodic memory is further dissected in [7].

The architecture was intended to support Non-Player Characters (NPCs) in a Role Playing Game (RPG). NPCs would always be running, independent of user interaction, and should be able to reconstruct their personal history. Namely, they should be able to tell what has happened in a time period and where they think a certain object is.

In the architecture, each agent has a goal structure that consists of several AND-OR trees. The OR nodes are said to be tasks. Certain tasks have parameters. These parameters represent slots for world objects. AND nodes are sub-goals, with each root node representing a high-level goal. There one high-level goal active at a time, and consequently only one tree is active at a time.

Concerning the memory part of the architecture, both short-term memory (STM) and long-term memory (LTM) are modeled. The authors name the representation of an object in LTM or STM as a phantom. A phantom has the following elements: state of the object, object's position, and time when the phantom was added to LTM.

STM storage is divided in 3 main sections: perception field (PF) that contains phantoms of objects currently perceived by the agent; own task field (TF) containing tasks that will be performed or are being performed; memory field (MF) that has phantoms retrieved from LTM. Concerning the first section, when an object is perceived by the agent, and its saliency value is high enough, a phantom of it is added to PF. Saliency values are calculated using a base saliency value, inherent to the object, and are higher if the object is linked with a task in the active goal tree. On the other hand, if an object is perceived, and its phantom is in MF, it is passed to PF. Considering the second section, tasks get to TF in two situations: if they are selected for execution by the action selection mechanism; or if an object that can be used in the task is perceived. Tasks are removed from TF once they are done or due to decay. The decay mechanism works in the following way: if a task is in TF but is not performed for a certain amount of time, it is removed from TF. Similarly, if a phantom is in PF or MF, and is not perceived, or used, for some time it is removed.

Turning now to the LTM, the storage structure is constituted of trees of tasks. Tasks are added to the LTM when they are removed from TF. Analogously, phantoms are added to LTM when they are removed from PF. Each task tree corresponds to a part of a tree, or the complete tree, existing in the goal structure. One difference between the two is that in LTM parameters are initialized with phantoms. The other is that sub-goals are not explicitly represented on the LTM trees: the trees in LTM are collapsed versions of the trees in the goal structure. When a task is added to the LTM, it is connected with the task in LTM that in the goal structure was its lowest ancestor (root at the top), and no link is added if there is no ancestor in LTM in these conditions. Tasks are stored with the following information: the time when the agent started to perform the task; the time when it ended executing it; and a goal, in case the task was a sub-task of a goal. Additionally, when a task is added to the LTM, the phantoms that are used as parameters for the task, are also added to LTM and connected to the task. LTM is schematically represented in Figure 3.1.

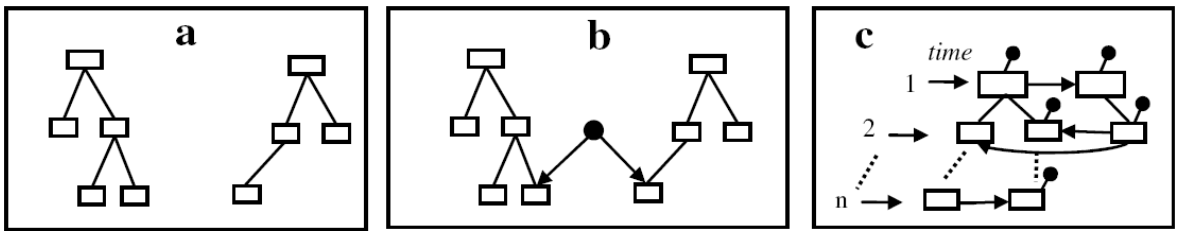


Figure 3.1: Full Episodic Memory LTM representation (from [7]). Rectangles represent tasks and dots represent phantoms

Tasks are removed from LTM due to a forgetting mechanism. Tasks with low importance are removed from the LTM when the agent enters a sleep state. Importance is calculated according to three factors: the layer in LTM tree (h), that is, the distance from the tree root; the number of days elapsed since the task ended (d); and the emotional relevance or saliency of the task. The importance expression is the following $e\frac{1}{h}\frac{1}{\sqrt{d}}$. Phantoms are removed when all tasks referencing them have been removed.

In storage there is a heavy emphasis on retrieval performance, with phantoms being hashed and tasks indexed by days. The retrieval process was supposed to enable the agent to textually

describe what has happened to him. Retrieval would be triggered due to a user question. Nevertheless such a system was not fully implemented in [7]. In the presented work, one can ask what the agent has done between two specified time points, at a certain memory layer level. The system returns a set of tasks on the LTM that were performed during the specified time period at the specified layer level. Lastly, authors found that the forgetting mechanism greatly influenced the storage space required by the system.

3.1.5 SALT – Schema-Associative Model of Memory

In the SALT (Schema-Associative Long Term Memory) agent memory architecture [5], long-term memory is organized in a network of nodes. Each node represents a mental model of some aspect of the world. Nodes have three main elements: propositions, header and activation level. Propositions are representations of information in first order predicate calculus. The header contains the concepts used in the propositions. Last, the activation value represents how easily can the information stored be retrieved. Information in long-term memory is searched to deal with perceived situations. When searching, the node whose header matches the situation, and has the highest activation value, is selected.

Delving into the maintenance of activation values, when a node is selected, its activation value increases. Furthermore, this increase is spread through the network. Nodes are connected to each other by weighted directed arcs. If a selected node X is connected to node Y (direction X to Y), node Y will be activated proportionally to the activation increase of X and to the weight of the arc. Recursively, activation of node Y will cause nodes connected to it to have their activation values increased. All these activation values decay exponentially with time.

The just described memory architecture was coupled with an affective appraisal mechanism in SALT & PEPPER [6]. We will not detail the affective system, yet a note should be given on the perspective of emotions taken. Instead of modelling full-fledged emotions (fear, joy, distress, etc), the authors model performance evaluators and attention-shift warnings, referring to them as emotion-signals. Performance evaluators consider the agent's performance in a given task, while attention-shift warnings enable the agent to shift attention, and cognitive resources, to more urgent or relevant aspects of the thought process.

These emotion-signals are part of the episodic memory retrieval process. Episodic memories are stored when a problem is solved, when the agent gathers information due to the world interacting with it, or when the agent makes a decision. They contain the time-frame in which storage took place and a reference to other memory nodes involved in the situation. Additionally, depending on the context, they may also contain: the problem dealt with, the solution conceived, the prospective decision outcome, and the origin of the obtained information.

Elicitation of these episodic memories can be caused by emotion-signals, as well as by external stimulus. When an emotion-signal is generated, are generated, they are matched against the header of all nodes in long-term memory (including episodic ones). The matching process is not specified, although the given example considers labels and valences: an emotion-signal matches a node if they have the same label and the same associated affective valence (positive or negative). As described before, the node that matches, and has the highest activation level, is selected.

The selected node is then increased in activation level proportionally to the emotion-signal's

intensity. If this node's activation value becomes higher than the activation value of the node currently being processed in the working memory, it is proposed to receive the agent's attention. Another mechanism, the interrupt manager, then decides if the current processing should be interrupted and the attention shifted to the elicited node. If the interrupt manager accepts the interruption, and the node is an episodic one, the decisions encoded can be re-used, or the information gathered put into focus. In this consists SALT & PEPPER's episodic memory retrieval process.

3.2 Emotion Architectures

As noted in Section 2.3, emotion is an essential element of agent believability [4][33]. If a reasoning system determines an agent's actions, it should consider emotions in order to make the agent believable. In this section we describe some of the agent architectures that model the influence of emotions on the action choice process. We focus on architectures that support appraisal because our model incorporates appraisal into the retrieval process. We will present the following architectures: the Affective Reasoner [16]; Émile [20]; the Interactive Pedagogical Drama architecture (IPD) developed for the project Carmen's Bright IDEAS [35]; the architecture used in the Mission Rehearsal Exercise (MRE) [21]; and FAtiMA (FearNot! Affective Mind Architecture) [13][14]. Many other architectures, supporting emotions, but not substantially focused on appraisal, have been developed [15][2][40].

3.2.1 Affective Reasoner

Clark Elliot developed the Affective Reasoner [16], a system for modelling emotions, reason about them, and act accordingly. The agent's goals, standards and preferences are encoded in a hierarchic system of frames. Each agent also keeps a model of other agents' goals, standards and preferences (concerns-of-others).

When an event occurs the system tries to match it to construal frames that encode how the agent interprets the world. If there is a positive match, the Affective Reasoner creates an Emotion Eliciting Condition Relation (EECR). More than one construal frame can have a positive match, so more than one EECR can be generated. A similar process occurs using the Concerns-of-Others (COO): events are matched against COO construal frames, and EECRs related with the fortune of other agents are generated. EECRs are generated from frames with a set of appraisal feature values. For each EECR, these values depend on the type of the construal frame from which it was generated. Two of these features are desirability-for-self and desirability-for-other. Desirability-for-self represents the extent to which an event enables, or hinders, the achievement of a personal goal. Desirability-for-other is the inferred desirability of an event for another individual.

After the EECRs have been created, domain independent rules, grounded on the OCC model [43], are used to generate emotions. Rules can generate conflicting emotions, or similar emotions due to different reasons. The approach taken here is to pass all emotion instances to the action generation layer. The idea is that agents can indeed have conflicting emotions, but they may not express all of them. Expression of emotions is dealt with in the action generation layer.

In the action generation layer actions vary from pure action tokens, to template actions, to simple plans. They can be divided in categories such as somatic responses, communicative verbal responses, behavioral responses directed towards and inanimate object, etc... For each generated emotion, an action is selected from each of the active action categories. If there are incompatible actions, one of them is chosen randomly. Actions are then expressed and constitute new simulation events. A final remark concerning the action generation layer: personality is expressed though activation and deactivation of action categories.

3.2.2 Émile

Émile's fundamental characteristic is that agents interpret events according to generalized rules grounded on their current plans and goals [20]. Appraisal of events is based on Clark Elliott's construal theory [16], but instead of appraising events in itself, it appraises the state of plans. It uses a small number of domain independent rules that reference a plan's structure, instead of a large number of domain-specific rules.

These rules set appraisal feature values that can be used to define EECRs. The overall appraisal of a plan is achieved by combining local plan appraisals. Quantification of appraisal intensity is done according to two variables: probability of goal attainment and goal importance. These are calculated by recursively analyzing plans sub-goals.

Given a series of appraisals it is necessary to integrate them into an emotional state. Each appraisal contributes with its intensity to an emotional bucket according to its label (anger, joy, fear, etc). This intensity contribute is decayed over time. The values in one emotional bucket may excite or inhibit values in other buckets. Each emotion will have an associated intensity by the end of this step. This intensities define the emotional state. The emotional state of an agent can influence the way he communicates and expresses himself to others. Emotion can also be used as a focus director when choosing which parts of a plan to explore first.

To conclude, the main arguments stated for choosing a plan based approach are: possibility to reason about the future, crucial in dealing with expectation emotions; easier identification of interactions between plans; generality of the appraisal rules; and the possibility to integrate plan oriented techniques into an emotion architecture. Unfortunately plan based approaches still have problems of efficiency and do not deal well with non-cognitive modulation of emotional state.

3.2.3 IPD – Interactive Pedagogical Drama

Marsella et al. developed an interactive pedagogical application called Carmen's Bright IDEAS [35]. Carmen's Bright IDEAS has an interactive story in which its characters are intelligent agents: Carmen, a mother with a son suffering of leukemia; Gina, a psychologist trying to help Carmen. Gina influences Carmen through strategy suggestions.

The applications architecture (IPD) uses a situation space model of multiple layers [34]. Each layer can abstractly be seen as finite automaton. Dialog acts can trigger transitions between states. There are four layers for each agent: problem solving, dialog model, physical focus and emotional appraisal. The problem solving layer is directly related with agent goals and models its techniques to achieve them. The dialog model defines in which state the conversation is in

from the perspective of the agent. Bellow we explain in more detail the emotional appraisal and physical layers.

Humans tend to express their emotions through non-verbal behavior. In a system were different emotions are active, there are conflicting non-verbal emotional expression tendencies. If the agent displays incoherent gestures, expressions, according to these conflicting emotions, the behavior will be perceived as incoherent. IPD deals with this problem through the physical focus layer. The agent can be in one of four physical focus modes. Each behavior can be consistent with one or more modes. Transitions between physical focus modes occur due to emotion state changes. Modes represent different degrees of self-centered to outward expression of emotions: strong-body focus, body-focus, transitional and communicative.

Let us now go deeper into emotional layer. It influences all other situation state layers. The emotion state highly depends on the appraisal of events. Appraisal is influenced by cognitive processes and feedback from action's consequences. Appraisal is a two-step process: in the first step the event is analyzed in terms of its relevance and congruency with the agent's motivations; in the second step it is analyzed in terms of four characterizing appraisal factors. These four factors are: accountability; expectancy; problem-directed coping potential; emotion-directed coping potential. Motivational congruency and these four factors mentioned have an associated value that is set based on appraisal rules. Whenever an event occurs it is matched against these rules and the corresponding appraisal, or appraisals, is generated with the specified values.

The system supports simulated memories of previous appraisals. For every type of problem the agent faces the set of appraisal values previously generated is stored. When the problem arises again, the appraisal values are reset to their previous values or an average is made with the current ones.

3.2.4 MRE – Mission Rehearsal Exercise

The Mission Rehearsal Exercise [21] is an interactive immersive learning system aimed at simulating real-world military scenarios in mission-oriented exercises. The user controls an avatar and all other virtual humans are intelligent agents. The believable agent framework used draws inspiration from Émile [20] and the IPD system [35]. It is domain-independent and addresses mainly non-deliberative emotions.

In what concerns emotion expression through behavior it uses the technique Physical Focus from IPD. Transitions between modes occur due to emotion states variations derived from an appraisal model similar to Émile's. Rules map the current emotional state into a specific focus mode. The focus mode influences cognition indirectly by changing the way the agent perceives the world around it, serving as a filter to external stimuli. Choice between alternative strategies available to achieve a goal might also be directly influenced by the focus mode. A system of anxiety was created in order to support physical focus. Anxiety is presented as correlated with non-specific threats to goals.

The choice of behavior at a given moment takes into account, besides the physical focus and the perceived world state, the current active behavior. Each agent analyses a set of possible plans and chooses the one that has the best match to the current situation. The plans are defined by sub-tasks, order between sub-tasks and who is responsible for the execution of each

one. It is possible to define that certain agents are concerned with the truth maintenance of specified conditions. Consequently these agents choose actions in order to assure that those conditions are verified.

3.2.5 FAtiMA – FearNot! Affective Mind Architecture

FAtiMA is an agent architecture developed alongside the application FearNot! [13][14]. FearNot! was created with the objective of addressing the bullying problem. In this application users interact with a character in a non-scripted story about bullying. The characters of the story are autonomous agents. Agents have goals they wish to achieve and standards of conduct they want to uphold.

FAtiMA generates agent's emotions due to reactive and cognitive reasons (Figure 3.2). The first step of the reactive appraisal is the use of emotional reaction rules to define appraisal factor values. These emotional reaction rules are similar to the ones presented in [36]. Emotion rules are structured in a hierarchy of three levels, each one corresponding to an attribute: subject, who did the action involved in the event; action, the action type; target, who or what was the target of the action. Events are matched against the emotion rules. The most specific emotion rule is chosen. Following the OCC model [43] an emotion is generated based on the generated appraisal values.

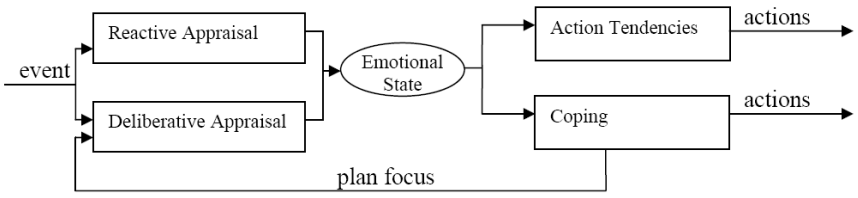


Figure 3.2: FAtiMA's emotion generation and emotion expression (from [13])

On the other hand, the cognitive process of emotion generation is based on plan analysis and goal expectations. The agent has a series of intentions, only one of which can be active at a certain time. Each intention can have a series of possible plans connected with it. When a plan comes into focus, the probability of it being achieved, or not, generates expectancy related emotions: fear and hope.

As the emotion generation is a continuous process, more than one emotion can be active at the same time. Each active emotion has an associated intensity. For emotions created reactively, appraisal variables define the emotion's base intensity. For cognitive emotions this value takes into account a plans's probability of success and its importance. Two other factors contribute in both cases for an emotion's intensity: mood and emotion threshold. Mood is a value that defines the overall valence of the agent's state, positive or negative. If an agent is in a good mood, positive emotions will be favored and negative ones lessened in intensity. A negative mood has the opposite effect. Reversely, positive emotions will increase the mood value and negative ones will decrease it. An emotion threshold defines a minimum value an emotion has to have in order to be activated. This value is subtracted to the emotion intensity. Thresholds are agent specific and emotion specific. They can be seen as the resistance an agent has to a

certain emotion, and be used to model personality. Finally, emotion intensities decay with time following the ideas proposed by Picard [46]. When an emotion reaches a value near zero it is removed from the active emotions set.

Emotions affect behavior through a reactive and deliberative module. In the reactive module behavior is defined according to action tendencies. These tendencies are encoded as action rules. Each action rule has an action, preconditions, and causing emotion characteristics. Emotion characteristics are: type of emotion, intensity, and features of the event causing the emotion. Emotions and their causing events are matched against all active rules (the ones whose preconditions are verified). The matched rule for which the causing emotion has the highest intensity is selected. The selected rule's action is executed.

In parallel with the reactive behavior generation, the cognitive module influences behavior through coping. Similarly to MRE [21], coping might involve problem-focused plan changing, or emotion focused world interpretation biasing. The intention selection is based on the associated prospect emotion intensity. Remember that these prospect emotions can either be fear or hope. The intention with the highest emotion intensity is selected. In this way emotions drive planning focus. So how can agent deal with intentions' corresponding emotions? Possible problem focused-coping strategies are adding sub-plans to a plan in order to eliminate open preconditions, or simply dropping a plan due to its low probability. Emotion focused strategies can influence the agent's interpretation of the world (e.g. lowering a goal threat probability).

3.3 Comparative Analysis of Architectures

Agent architectures that support autobiographic memories have been investigated in several previous works, however modelling episodic memory retrieval for believable agents is not an extensively debated topic. We analyzed to which extent several architectures support believable autonomous agents that remember episodic memories that were emotionally relevant to them. Additionally, we kept in mind that our approach to this problem was to model ecphory (with retrieval cues) and recollective experience (with re-appraisal of past events).

Ho, Dautenhahn and Nehaniv have developed several autobiographic memory architectures in the context of artificial life [24]. In them, memories are paths to resources, and storage is triggered by a timer or by an event. Retrieval happens when a resource is needed and the retrieval process consists of reconstructing previously walked paths. Although the model supports retrieval of personal information (the agent's paths), this retrieval is not an emotional experience. Moreover, the work focuses much more on the agent's survival skills, than on believability characteristics.

In FAtiMA [14] there is a greater concern with believability, with emotion and personality concepts having an important role. The system supports both emotional appraisal and autobiographic memories. Furthermore, these memories contain the agent's emotional reaction to the events. Nevertheless, memory retrieval is not presented as an emotional appraisal process: there is no real recollective experience. Memories are simply retrieved when the agent wishes to summarize its life story.

Other architectures that support appraisal in agents were analyzed. They share the appraisal theory inspiration, having different ways of dealing with the agent's situation interpretation,

emotion intensities and behavior modulation due to emotions. However, in the Affective Reasoner [16], Émile [20] and MRE [21], the appraisal system is not used in a memory retrieval process. IPD [35] simulates memories by caching appraisal values for each problem the agent deals with. Although this may be true, the concepts of ecphory and retrieval cues are not really present.

In [26] memories are combined and encode both goals and event coping strategies. Autobiographic memories are used for extracting goals, verifying if they have been achieved and choosing a reaction strategy to an event. The model is, however, more directed to the semantic part of autobiographic memory, than to episodic memory.

Brom and Lukavsky [8] proposed an agent architecture that supports episodic memory retrieval. In long-term memory, memories are structured as trees of performed tasks. These tasks are removed from long-term memory according to a forgetting mechanism that takes into account, among other things, the time passed since the task was performed. Additionally short-term memory is also modeled. Despite all its features, memory retrieval is described as a data base process, and not as an emotional experience. There is a proposal of using retrieved data for reconstructing a personal history, however this part of the system was not implemented.

The SALT architecture [5] integrated in the SALT & PEPPER architecture [6] also supports episodic memories. In it, memories are organized as a network of nodes, with different activation values. The most activated nodes are more susceptible to be retrieved. A matching process between stimulus and nodes is referenced in the retrieval process, which might be interpreted as echory. Nevertheless, a clear generic definition of ecphory is not presented. Furthermore, by not directly supporting full-fledged emotions, its applicability in modelling clearly identifiable emotions is hampered, thus limiting its use for believable agents [4].

In Table 3.1 we present a comparison of all described architectures[1]. On the whole, each architecture only fulfills part of our requirements. Modelling ecphory and modelling recollective experience are relatively unexplored subjects in the analyzed work. We will explore them in the next chapter.

[1]Note three of IPD's attributes have a "*" because memory is only simulated in this architecture. Additionally, in SALT & PEPPER ecphory is referenced although not clearly defined.

Table 3.1: Agent Architectures Comparison

architecture	Autobiographic or Episodic memories	Emotional Memories	Believability Focus	Emotional Appraisal	Ecphory	Recollective Experience
Artificial Life	yes	no	no	no	no	no
FAtiMA	yes	yes	yes	yes	no	no
Émile	no	no	yes	yes	no	no
MRE	no	no	yes	yes	no	no
IPD	yes*	yes*	yes	yes	no	yes*
Discrepancy	yes	no	no	no	no	no
Affective Reasoner	no	no	yes	yes	no	no
Full Episodic Memory	yes	no	no	no	no	no
SALT & PEPPER	yes	yes	no	yes	yes*	no

Chapter 4

Model

In order to build agents that exhibit emotional behavior which is influenced by its episodic memories, we have developed a model for agent episodic memory retrieval (see Fig. 4.1). Retrieval is a two step process, as has been proposed for human memory [54]: ecphory and recollective experience. In ecphory, episodic memory traces are correlated with retrieval cues. The memory traces that have a high correlation with retrieval cues are selected and fed to recollective experience. In recollective experience the memory traces' events are re-appraised resulting in a possible change in emotional state.

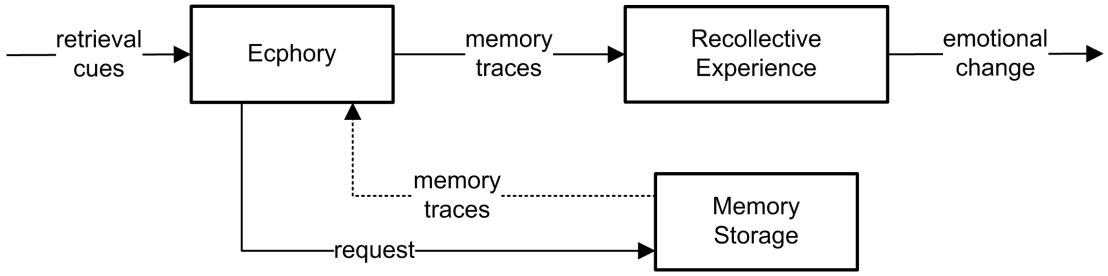


Figure 4.1: Episodic memory retrieval

4.1 Ecphory

As mentioned above, in ecphory, retrieval cues are correlated with episodic memory traces. Retrieval cues are stimulus that can either be external (e.g. smell) or internal (e.g. mood value). We describe a group of retrieval cues as a point in a perception feature space (in analogy with Attneave's work [1]). Considering a perception feature space S, a group of retrieval cues can be described as an $r \in S$. A perception feature, or dimension, will typically result from combining several retrieval cues (Figure 4.2a). For example, several visual stimuli may result in an overall perceived light intensity feature value. On the contrary, categories of stimulus may need more than one dimension to be described (Figure 4.2b). For example, several features can define auditive perception, each one representing the sound intensity perceived at a certain frequency range. Additionally, a perception feature may be mapped to a single stimuli (Figure 4.2c). As an example, a mood value may be simply mapped to a mood feature. Lastly, a representation of a group of retrieval cues may result from different types of mappings (e.g. the group of retrieval

cues r', defined in a seven dimensional perception feature space, may be represented by the tuple $(l1, f1, f2, f3, f4, f5, m1)$, in which $l1$ corresponds to an overall light value, $f1$ to $f5$ are intensities for different frequencies, and $m1$ is a quantification of mood).

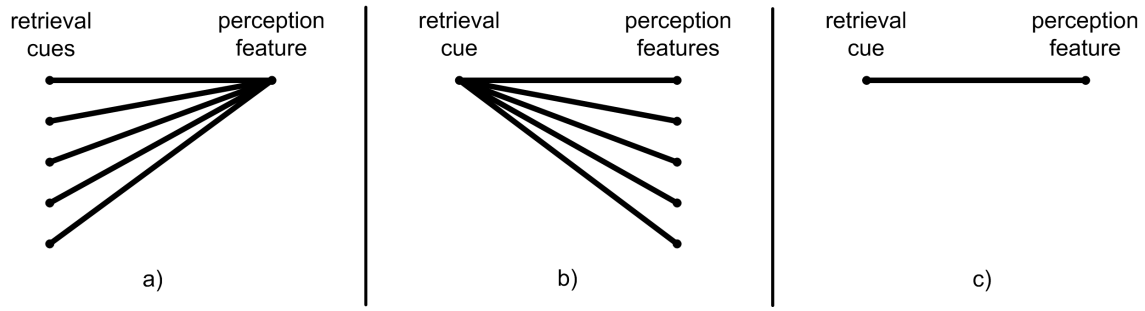


Figure 4.2: Retrieval cues mapped to perception features - a) many retrieval cues map to one perception feature b) one retrieval cue maps to several perception features c) one retrieval cue maps to one perception feature.

A group of retrieval cues is a point in a perception space, and we wish to be able correlate it with memory traces. Consequently, we define a memory trace m as a tuple in which one of its elements, the perceptual point, is a point in the same perception space ($p_m \in S$). Other elements of the memory trace will be described further on.

We correlate a memory trace m with a group of retrieval cues r, by calculating a distance between r and p_m using a function $d : S \times S \rightarrow \Re^+$. We will refer to such a distance between points in a perception space as *perceptual distance*[1]. If the perceptual distance between a memory trace m and the retrieval cues is small, then m is selected for recollective experience.

Santini and Jane [51] have analyzed several functions to calculate perceptual distances. Two of these functions are the euclidean distance and the city-block distance. If we consider r_i to be the feature number i of a group of retrieval cues r, p_{m_i} to be the feature number i of the perceptual point p_m, and S to have n dimensions, the euclidean distance between r and p_m would be given by the expression $\sqrt{\sum_{i=1}^{n}(r_i - p_{m_i})^2}$, and the city-block distance by $\sum_{i=1}^{n}|r_i - p_{m_i}|$. Both are Minkowski r-metrics with $r = 2$ and $r = 1$ respectively. The general expression for a Minkowski r-metric distance would be $[\sum_{i=1}^{n}|r_i - p_{m_i}|^r]^{\frac{1}{r}}$.

Thus, the two main parameters of ecphory are:

- The perception feature space S, including which perceptual dimensions will be used and how many.
- The perceptual distance function $d : S \times S \rightarrow \Re^+$.

4.2 Location Ecphory

After describing how a generic ecphory would work in an agent architecture, we shall now pursue a more specific approach. Our Ecphory model is motivated by the idea that humans retrieval process results from the interaction between memory traces and retrieval cues [49]. If a person is

[1]It can be argued that the activation of certain memories should depend on how long they have been stored. However, we believe that such a phenomenon can be better mapped to a forgetting mechanism, such as the presented in [8], than to filtering memory traces during ecphory.

exposed to stimuli similar to the ones he, or she, was exposed during the occurrence of an event that is stored in memory, these stimuli can act as retrieval cues for that memory. If a person passes by a location where an event took place, an event he, or she, has an episodic memory of, and that location has not changed dramatically, there is a possibility of exposure to similar stimuli.

Consider the following situation: an individual A passing by the spot where she was first kissed. As individual A passes by, she might smell the sent of near flowers, be again exposed to the colors of the garden, gaze at the mountain landscape, feel the crunchy texture of the ground. All of these external stimuli act as retrieval cues, and individual A remembers her first kiss. Note that the mentioned stimuli are perceived in the garden. Hence, instead of saying that the individual stimuli elicit the kissing memory, one can say that the garden's stimuli elicited the episodic memory. In the end, *the garden's location is acting as an indirect retrieval cue for the memory.*

Of course if the garden had been bulldozed, replaced by a parking lot, and the view was now hidden by a shopping mall, the retrieval cues would be absent, and consequently the location could hardly be seen as an indirect retrieval cue for the episodic memory. Thus, the exposed situation as a whole shows that *locations can be interpreted as indirect memory retrieval cues when they have not changed dramatically.*

If we translate this intuition into our agent model of ecphory:

- the perception feature space (S_n) can be defined as the agent's physical space (e.g. 3D physical space).
- the perceptual distance function (d) can be defined as the euclidean distance between the location where the past event occurred, and the location where the agent currently is.

The result is that the closer the agent is to the past event's location, the higher the probability of retrieval. We name this specific ecphory as location ecphory. It is a simplification of the generic ecphory: on one hand it replaces direct stimuli input by physical locations; on the other hand, it only accounts for retrieval of a memory trace when passing by the location where the memory trace's past event took place.

For instance, consider that an agent has a memory trace m about an event that took place in a location $l1$. Now take in account that the agent is passing by a location $l2$ physically distant from $l1$, but quite similar in terms of appearance. As the two locations are similar, the stimuli that the agent was exposed to when storing the memory trace m is similar to the stimuli it is currently perceiving. Consequently, The currently perceived stimuli, acting as retrieval cues, should in principle have a high correlation with memory trace m. This high correlation would lead the memory trace to be selected for recollective experience. However, location ecphory does not account for this situation because $l2$ and $l1$ are physically distant, which will result in a low correlation value.

In spite of its limitations, from an engineering perspective, location ecphory is much less demanding on the sensor detail of a synthetic autonomous agent. Agents just need to be able to approximate their current physical location. They do not need to have a wide range of simulated sensors covering smell, sights, sounds, colours, etc. The ability to approximate a current physical

location is much more common in agents than detailed simulated perception. Therefore we believe that location ecphory can be integrated into a wider range of agent architectures than a more generic ecphory model.

4.3 Recollective Experience

After the traces are selected by location ecphory, or another ecphory parametrization, there still needs to be a recollective experience. According to Tulving [53] episodic memories allow humans to relive past experiences, so the recollective experience should have similarities with the original experiencing process. If an agent is to experience an event by appraisal, when a memory trace is retrieved, the event that is linked to it should be appraised a second time. Hence, the recollective experience will essentially be an appraisal experience.

Before we further describe the recollective experience, we need to define the concept of emotional reaction, emotion and emotional state. These definitions are inspired in the OCC model [43] and on FAtiMA [13]. We start by laying down a background scenario that will serve to exemplify the emotion definitions.

Two agents (meemo 1 and meemo 2) are moving in a tunnel. Meemo 1 and meemo 2 are friends. Meemo 1 witnesses meemo 2 falling in a deadly trap. Meemo 1 evaluates this event as undesirable for meemo 2 and also as undesirable for itself (as meemo 2 was its friend). Meemo 1 will have an emotional reaction to the event.

In our model an *emotional reaction* is a quantified evaluation of an event, defined by a pair $\langle AV, E \rangle$ in which:

- AV contains the set of appraisal values, two of which are desirability-for-self and desirability-for-other. Each appraisal variable represents an evaluation of the event through a specific perspective of the agent. Desirability-for-self represents the extent to which an event enables, or hinders, the achievement of a personal goal. Desirability-for-other is the inferred desirability of an event for another individual [16]. In our example, meemo 1 might have as a goal "stay alive" which will lead to a low value of desirability-for-self. Additionally meemo 1 can have a goal "meemo 2 stay alive" which leads to an even lower value of desirability-for-other.

- E specifies the event that generated the reaction. In the example, this element might have information such as "meemo 2 fell in trap located in tunnel on spot b3". We use the term event as a generalization of the OCC's appraisal evaluation focus: on consequences of an event, on the agency element of an event, or on an object of an event.

We define *emotion* as a valanced evaluation of an event described as a 4-tuple $\langle E, ET, EI, V \rangle$ in which:

- E contains information about the event that elicited the emotion (e.g."meemo 2 fell in trap located in tunnel on spot b3").

- ET specifies the emotion type according to the OCC model [43] (e.g. pity).

- EI specifies the current intensity scalar value (non-negative).

- V specifies the valence of the emotion (positive or negative). The valence is directly dependent on the emotion type. For example, joy emotions are positively valanced and pity emotions are negatively valanced.

An *emotional state* is defined by a 2-tuple $\langle AE, M \rangle$ in which:

- AE contains the set of emotions the agent is currently feeling.
- M specifies the mood value. Mood is a bounded scalar value that represents the agent's overall emotional state valence. Low values represent a bad mood and high values represent a good mood. For example, meemo 1 learns how to detect traps, causing it to feel joy, and in turn rising its mood. Shortly afterwards it detects a trap and feels pride, causing its mood to rise even higher.

We can now proceed with the model's description. The recollective experience process flow has three main steps:

1. Generating emotional reactions from events.
2. Generating emotions from emotional reactions.
3. Integrating generated emotions into the emotional state.

Extensive work has been done regarding all these steps, being FAtiMA [13] and Émile [20] examples of this. For the recollective experience one just needs to use a model such as the just mentioned. The past event information is extracted from the selected memory trace and then this information is fed into a generic appraisal module. We will use the term *retrieval event* to refer to a past event that will be re-appraised.

Additionally, our model ties in with the OCC model [43] that is a grounding theory for many agent appraisal models [13][20][16]. The OCC model specifically refers that appraised events can be in the recent or remote past (pg.86). Moreover, one of the presented factors that should influence emotion intensity is psychological proximity (pg.62). One can interpret the perceptual distance used in ecphory as being inversely related to psychological proximity. Only memory traces that are perceptually close to perceived stimuli (the distance is small) are selected. As a result, memory traces that are perceptually far from the current perceived stimuli are not re-appraised. With this we simulate an appraisal that has a very small effect on the emotional state, too small to count.

However, if we consider a generic appraisal module, some modifications need to be made. Following the view that a person can relive a past event as an observer or as an actor [38], agents will be able to do the same. Different architectures of appraisal use different structures for creating emotional reactions (construal frames, plans, reactive rules, etc), and these structures can change over time. When re-appraising an event the agent will be able to evaluate it according to its current evaluation structures (as an observer of its "past-self"), or use the emotional reaction to the event when it first occurred (as an actor in the event). Due to this last point, memory traces will also have the emotional reaction the agent had to the past event.

After emotional reactions to events have been created (*step 1*), they can be used to generate emotions (*step 2*). In a generic appraisal module, the only change that needs to be made, is

to decrease the intensity of emotions, or of potential emotions, when they are generated by re-appraisal of past events. With this decrease we try to encode the idea that memory retrieval is, in general, a less intense experience than the original one [38]. *Step 3* of a generic appraisal system does not need to be modified when the system is used to create a recollective experience.

4.4 Memory Storage

In general, each emotion that was successfully generated is passed to memory storage, together with the event that caused the emotion. Choosing to store emotion eliciting events is supported by research stating that in humans emotions drive event focus and consolidation [45], and that emotion arousal extends the durability of memories [38]. However, if the emotion was generated due to a retrieval event, no memory trace is stored. This choice was made to avoid recursive memory retrieval.

Memory storage creates an episodic memory trace as a 5-tuple $\langle Pp, D, T, Er, Em \rangle$ in which:

- Pp specifies a point in the chosen perception space (S) representing the perceptions the agent was exposed to when appraising the event. This point will be a physical location if a location ecphory is used (e.g. in tunnel location b4).
- D contains a description of the event (e.g. companion fell in trap).
- T defines the time stamp when the event started.
- Er specifies the emotion reaction to the event.
- Em specifies the emotion elicited by the appraisal of the event.

Memory traces are initially stored in a short-term memory storage (STM), and after a few seconds are passed to the long-term memory storage (LTM). It should be noticed, that only events that elicit emotions are stored at all, hence we filter memory traces before they go to STM. Abstractly, we are representing that events not eliciting emotions initially go to the STM, but are eventually discarded dur to their lack of emotional relevance.

Additionally, when a memory trace is selected by ecphory, it passes from LTM to STM. Retrieval abstractly represents passing memories from long-term memory to short-term memory. Consequently, if they are already in short-term memory, they should not be retrieved. Hence ecphory only selects memory traces that are in LTM, and ignores memory traces in STM.

As a final remark it should be noted that no model for memory forgetting will be presented. Our research focus is on episodic memory retrieval, consequently only the memory storage elements strictly relevant to the retrieval process are described. Nonetheless, a forgetting mechanism similar to the one presented in [8] (see Section 3.1) could be easily defined. The main difference would be that the importance expression would only take in account emotion relevance and time elapsed.

4.5 Summary

In this chapter we described a model for agent episodic memory retrieval. In our retrieval model one starts by calculating a perceptual distance between the perceived stimuli (retrieval cues) and the perceptual memory of each episodic memory trace. This process is named ecphory. We presented an approximation of generic ecphory that is grounded on the idea that locations can be interpreted as indirect retrieval cues (location ecphory). After ecphory, memory traces for which the calculated perceptual distance is small are selected to be re-experienced (recollective experience). We described how a generic appraisal model could be adapted to serve as a recollective experience model. Namely, that past events could be re-evaluated according to the agent's current evaluation structures (reliving as an observer), or the original emotional reaction to the event could be re-used (reliving as an actor). Finally we formally defined the memory trace concept and described how memory storage was divided in short and long-term memory.

Chapter 5

Implementation

Having defined a model for episodic memory retrieval in the previous chapter we will now describe how it was implemented. We start by giving an overall overview of the agent architecture. Following this, we examine the different modules that compose the architecture. We proceed by describing the application in which the agent architecture was integrated, together with a specific module for the Behavior of agents. Last, we present a complete example of the working system and some closing remarks.

5.1 Agent Architecture

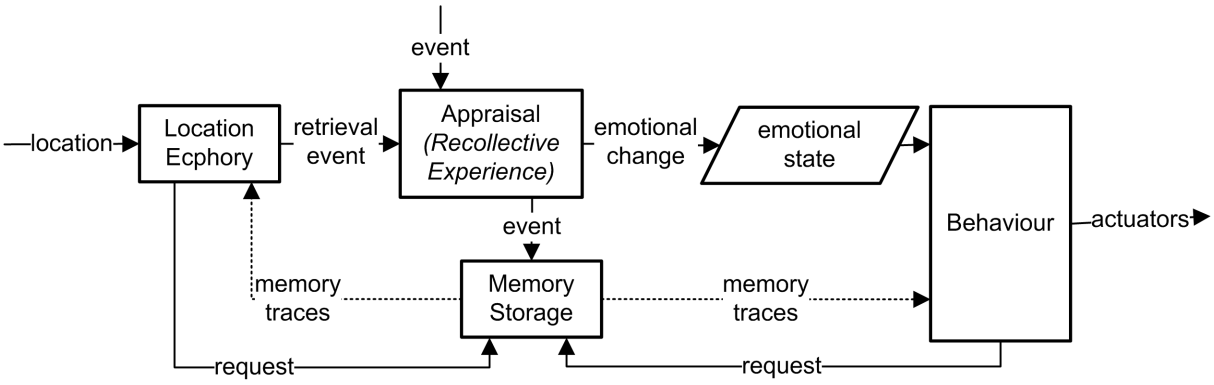


Figure 5.1: Agent Architecture

Before we go through the implementation's modules (written in C++), we will present a simple overview of the agent architecture (see Figure 5.1). The *Location Ecphory* module is responsible for constantly trying to match the agent's current location with stored memory traces. If there is a match, the memory trace's event is fed into the *Appraisal* as a retrieval event. The Appraisal acts as a *Recollective Experience* enabling retrieval events to be re-experienced. In parallel, non-retrieval events (present events), when generated, are also fed into Appraisal. All events are appraised and, as a consequence the emotional state may be changed. If the emotional state is changed due to a non-retrieval event, the generated emotions and their causing events are stored as memory traces in *Memory Storage*. Meanwhile the *Behavior* uses the emotional state, and memory traces from Memory Storage, to determine which actuators should be activated.

5.1.1 Events

In the architecture's overview, we mentioned that all events are fed to the Appraisal. These events are generated either by sensors (non-retrieval events) or by the Location Ecphory (retrieval events). Non-retrieval events have the three parameters indicated in Table 5.1. Retrieval events will be further described in a later subsection.

Table 5.1: Non-Retrieval Event Parameters

Parameter	Description
type	Enumerate representing the type of the event. In the application it is assumed that there is a finite number of event types.
location set	Boolean that is true if and only if the *location* parameter is set.
location	If the event took place at a specific point in space, it will have the world coordinates of that point (e.g. if an agent finds a raspberry bush, the location of this event could be the exact coordinates of the bush). If however, the event's action is spread trough an area, the location will be the world coordinates of a point representing the event's action center (e.g. if an agent performs a dance in an area, the location of this event can be the centroid of this area).

Non-retrieval events will typically have the location set parameter defined as *true*, except for events in reaction rules (described in the Appraisal section). Additionally there is another special type of events called *witness events*. Witness events are events that the agent witnesses, but in which it is not directly involved. In fact, in witness events the agent is not an agency element of the event, that is, the agent's actions are not directly causing the event. Witness events have type *EventWitness*, have location set as *false* and also contain an additional parameter (*witnessed event*). This parameter represents the event being witnessed by the agent. We show the textual representation of four non-retrieval events, the last of which is a witness event.

```
EventFindRaspberryBush[l(C1: 450,C2: 300)]
EventDance[l(C1: 50,C2: 400)]
EventBitBySnake[l(C1: 450,C2: 150)]
EventWitness - EventReachExit[l(C1: 200,C2: 50)]
```

This last event indicates that the agent witnessed another agent reaching the exit that was at coordinates (200,50). The third event indicates that the agent was bit by a snake at coordinates (450,150). Events, such as the just presented, can elicit emotions in the agent (process that will be described in Section 5.1.5). Events and caused emotions are stored together as memory traces.

5.1.2 Memory Traces

A memory trace has only three parameters as presented in Table 5.2. It is noticeable that two elements of the memory trace concept (described in Chapter 4) were not translated into parameters: the perceptual point and the emotion reaction. The perceptual point is not present because we take a location ecphory approach and use the event's location instead. The emotion

reaction is not present because we only implemented the recollective experience as an observer, hence the emotion reaction is not necessary in this case[1].

Table 5.2: Memory Trace Parameters

Parameter	Description
event	Event which the memory is about.
emotion	Emotion caused by the event.
time stamp	It starts by being the time stamp when the event started, but it can change. Details will be presented in the Memory Storage section.

We present the textual representation of two memory traces. The emotions will not be represented as their parametrization will be described later on.

```
(MemoryTrace:
   _event: EventFallHole[l(C1: 150,C2: 350)]
   _causedEmotion: ...
   _timestamp: 65)
(MemoryTrace:
   _event: EventReachExit[l(C1: 200,C2: 50)]
   _causedEmotion: ...
   _timestamp: 1034)
```

The first memory trace would indicate that the agent fell in a hole at coordinates (150,350). The event would have occurred 65 seconds after the start of the simulation. The second memory trace would indicate that the agent had reached an exit 1034 seconds after the start of the simulation, and that this exit was placed at coordinates (200,50).

5.1.3 Memory Storage

All memory traces are stored in *general memory traces* set. However, we conceptually separate memory traces in long-term memory (LTM) from memory traces in short-term memory (STM). A memory trace is considered to be in STM if one of these two mutually exclusive conditions are verified:

- The memory trace is present in general memory traces and the difference between its time stamp and the current time is smaller than *short term memory duration*: $CurrentTime() - TimeStamp(memory\ trace) < short\ term\ memory\ duration$. Short term memory duration can be parameterized in a configuration file.
- A copy of the memory trace is in the *partial short term memory traces* set (albeit with a different time stamp).

All other memory traces, that do not verify either of these conditions, are considered to be in LTM. When a memory trace is selected by Location Ecphory a copy of it is added to partial short term memory traces. This copy's time stamp is set to the instant in which it was added to partial short term memory traces.

[1] We discuss this choice in Section 5.1.5

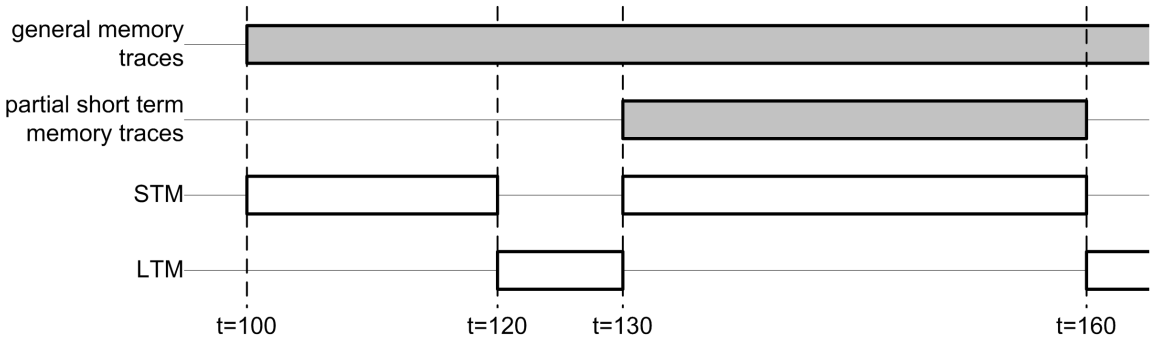


Figure 5.2: Memory Storage Example - The shaded bars represent the presence of a memory trace in general memory traces and in partial short term memory traces. The white bars indicate if the same memory trace should be considered to be in short-term memory (STM) or in long-term memory (LTM).

Memory traces in the partial short term memory traces set are progressively removed. This removal is done trough a lazy update: only when a memory trace is going to be checked if it belongs to the set, is the set updated. The update removes all memory traces whose time stamp differs short term memory duration, or more, from the current time.

$$CurrentTime() - TimeStamp(memory\ trace) \geq short\ term\ memory\ duration$$

Consider the following example in which the short term memory duration was set to 20 seconds. All time stamps will be presented in seconds and the situation is schematically represented in Figure 5.2. An agent $a1$ stores a memory trace $m1$ in Memory Storage at time t=100 ($TimeStamp(m1) = 100$). $m1$ is added to the general memory traces set. From t=100 to t=119 $m1$ is considered to be in STM because $CurrentTime() - TimeStamp(m1) < 20$. From time t=120 on, $m1$ is considered to be in LTM because $CurrentTime() - TimeStamp(m1) \geq 20$ and $m1$ has not been added to partial short term memory traces. At time t=130 $m1$ is selected by Location Ecphory to be re-experienced. Consequently a copy of $m1$, that we will refer to as $m1'$, is added to partial short term memory traces. Memory trace $m1'$ has been added to partial short term memory traces at t=130 therefore $TimeStamp(m1') = 130$. From t=130 to t=159 $m1$ is considered to be in STM because its copy ($m1'$) is in partial short term memory traces. At t=160 the system checks if another memory trace is in LTM. This causes a lazy update that will remove $m1'$ from partial short term memory traces because $CurrentTime() - TimeStamp(m1') \geq 20$. As a result $m1$ is again considered to be in LTM because $CurrentTime() - TimeStamp(m1) \geq 20$ and there is no copy of $m1$ in partial short term memory traces.

5.1.4 Location Ecphory

As mentioned in the architecture's overview, Location Ecphory is responsible for selecting which memory traces should be re-experienced. We followed closely the location ecphory approach described in the previous chapter. At each time step, all memory traces in the general memory traces set are matched against the agent's current location. If the euclidean distance between

the agent's current location, and the memory trace's event location[2], is smaller than *location ecphory distance* (parameterizable in a configuration file), there is an ecphory match.

$$d_{rm} = euclideanDistance(Location(agent), Location(Event(memory\ trace)))$$

$$d_{rm} < location\ ecphory\ distance$$

Furthermore, we only select memory traces for re-appraisal if, besides having an ecphory match, they are in fact in LTM. Consequently, when an agent is in the close proximity of a location where an event took place, and that event is stored in the agent's LTM (through a memory trace), memory retrieval of that event is triggered. In this process, more than one memory trace may be selected, because several memories can be linked with past events that occurred close to where the agent is. For each memory trace that was selected a retrieval event is created.

Besides the parameters previously presented for non-retrieval events, retrieval events have an additional one: retrieved event. The retrieved event is the event parameter of the memory trace that was selected. Furthermore, the type parameter is set to "Retrieval" and both location set and location parameters are set to the values of the respective retrieved event parameters. Generated retrieval events are fed into the Recollective Experience (Appraisal). Additionally, matching memory traces are copied to the partial short term memory traces set (as described in the previous section).

Consider the following example in which the location ecphory distance was set to 5 meters (all distances will be presented in meters). An agent $a1$ has three memory traces in Memory Storage: $m1$, $m2$ and $m3$. Their events are respectively $e1$, $e2$ and $e3$, and these events locations are $l1$, $l2$ and $l3$. The locations are defined in a 2 dimensional space and have the following coordinates: $l1 = (7, 3)$, $l2 = (5, 3)$ and $l3 = (1, 1)$. The agent is currently at location $la1 = (7, 4)$. All locations are schematically represented in Figure 5.3. Additionally, we know that $m2$ and $m3$ are in LTM while $m1$ is STM. In this situation there would be an ecphoric match for $m1$ and another for $m2$ due to the fact that $l1$ and $l2$ are closer than 5 meters from $la1$. There would be no ecphoric match for $m3$ because $l3$ is further than 5 meters from $la1$. Since $m1$ and $m2$ had an ecphoric match, we check in which memory system (STM or LTM) they are. As only $m2$ is in LTM, only $m2$ will be selected for Recollective Experience. Hence a retrieval event re will be generated. Its retrieved event parameter will be $e2$, its location set will have the value *true*, and its location will be $l2$. Retrieval event re will then be fed into Appraisal. Meanwhile, as $m2$ was selected, it passes to STM, and consequently will not be able to be selected again for Recollective Experience for the duration of short term memory duration.

5.1.5 Appraisal - *Recollective Experience*

As previously mentioned, the Appraisal is used to evaluate present events as well as re-experience past ones. To develop it, we started by translating from the Java programming language to C++ the reactive appraisal part of FAtiMA's implementation [13], adapting it when necessary. For instance, we changed it so it would treat differently retrieval events and non-retrieval events.

[2]Apart from events in reaction rules (described in the Appraisal subsection) all other events have their location set

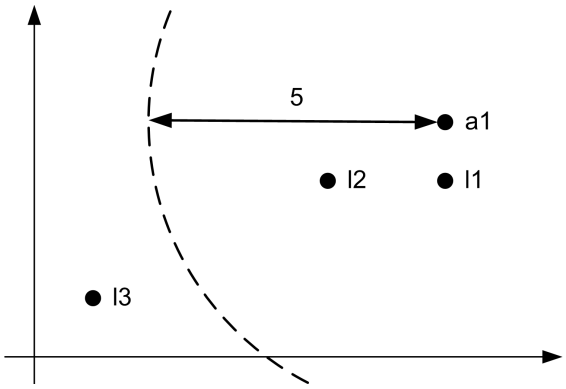


Figure 5.3: Location Ecphory Example

Several emotion connected concept classes (such as emotion, emotional state, etc.) were also translated from FAtiMA's implementation. To describe all the Appraisal's elements, we will follow the steps defined in the model for a generic Recollective Experience process flow (see Section 4.3).

Step 1 - Generating emotional reactions from events

Appraisal receives retrieval events from Location Ecphory, and non-retrieval events generated by sensors. It starts by using these events to produce emotional reactions. An emotional reaction has the parameters presented in Table 5.3.

Table 5.3: Emotional Reaction Parameters

Parameter	Description
desirability for self	Integer varying between -10 and 10 (except 0), or *null appraisal value* (integer not in this range). A negative value indicates that the event hinders the achievement of an agent's goal, and a positive value indicates that the event enables the achievement of an agent's goal. *null appraisal value* indicates that the event has no effect on the agent's goals.
desirability for other	Same as desirability for self but in regard to other agents' goals.
praiseworthiness	Integer varying between -10 and 10 (except 0), or *null appraisal value* (integer not in this range). A negative value indicates that the agency element in the event violates an agent's standard, and a positive value indicates that the agency element of the event upholds an agent's standard. *Null appraisal value* indicates that the event does not affect the agent's standards.
event	Event that caused the emotional reaction.

Emotional reactions are generated from events using reaction rules. A reaction rule has the same parameters as an emotional reaction, however its event does not have a defined location (location set = *false*). Each agent has a set of reaction rules that does not change during the simulation. Each received event is matched against all reaction rules of this set. Matching consists of comparing the event with the reaction rule's event. In turn, comparison between two events is done using a function whose result values vary between 0 (no match) and 10 (total match). Two events of different types have a comparison value of 0. Two events with all

parameter values equal have a comparison value of 10. In all other cases the comparison value is a positive smaller than 10.

When the comparison value between an event and the reaction rule's event is positive an emotional reaction is generated. This emotional reaction has the same parameter values for desirability for self, desirability for other and praiseworthiness as the reaction rule, and the event parameter is set to the event that was matched with the reaction rule. To illustrate this process we will describe a situation in which one event ($e1$) is matched against two reaction rules ($r1$ and $r2$). We start by showing a textual representation of the three.

```
e1 : EventFindRaspberryBush[l(C1: 450,C2: 300)]
r1 : [EmotionalReactionRule]
   _event: EventFindRaspberryBush,
   _desirabilityForSelf: 2,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
r2 : [EmotionalReactionRule]
   _event: EventFindMonster,
   _desirabilityForSelf: -5,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
```

When matching $e1$ and $r1$, $e1$ would be compared with the *Event Find Raspberry Bush*. The value for this comparison would be positive because both events have the same type (*Find Raspberry Bush*). Hence an emotional reaction would be generated with the following textual representation:

```
[EmotionalReaction]
   _event: EventFindRaspberryBush[l(C1: 450,C2: 300)]
   _desirabilityForSelf: 2,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
```

The desirability for self, desirability for other and praiseworthiness are the same as $r1$ and the event parameter is $e1$. Turning now to $r2$, when matching this rule with $e1$, $e1$ would be compared with the *Event Find Monster*. The value for this comparison would be 0 because the events have different types (*Find Raspberry Bush* and *Find Monster*). Therefore no emotional reaction is generated due to $r2$.

If we analyze the reaction rules of the example one could extrapolate two possible agent goals: "finding raspberry bushes"; "staying away from monsters". Alternatively one can think of more generic goals such as "finding food" and "staying away from danger", that in turn might be related to even more generic goals such as "feeding self" and "staying safe". Nonetheless the agent's goals are not represented explicitly in the architecture, but rather implicitly through the reaction rules, following the approach also taken in Martinho [36].

There are two exceptions to the generic matching process described, first of which concerns witness events. If a reaction rule's event is a witness event (type=*EventWitness*), and the event

to be matched is not of type *EventWitness*, the comparison value is, as would be expected, 0 (no match). However, if the event to be matched is of type *EventWitness* a second comparison must take place: the witnessed event parameter of the reaction rule's event must be compared with the witnessed event parameter of the event to be matched. Consider reaction rule $rw1$ and event $ew1$ whose textual descriptions are presented bellow.

```
ew1: EventWitness - EventReachExit[l(C1: 200,C2: 50)]
rw1: [EmotionalReactionRule]
   _event: EventWitness - EventReachExit,
   _desirabilityForSelf: 1,
   _desirabilityForOther: 5,
   _praiseworthiness: NULL_APPRAISAL_VALUE
```

In this case $rw1$'s event is an witness event and $ew1$ is also a witness event. Therefore, "EventReachExit[l(C1: 200,C2: 50)]" will be compared with "EventReachExit". As these two are of the same type, the result will be positive (smaller than 10 because the second event does not have a defined location). In such situations, the result obtained for the witness event parameters is used as the result for the matching between the reaction rule and the original event. Consequently the result of the comparison between $rw1$ and $ew1$ will be the previously obtained positive value.

The other matching exception regards reaction rules for retrieval events (that we will name *retrieval reaction rules*). The idea behind appraising retrieval events, as described in Chapter 4, is that by doing so the agent is able to relive past events, similarly to episodic memory retrieval in humans [53]. We have implemented this by creating an additional reaction rule for each reaction rule containing a non-retrieval event. The new reaction rule has the same desirability for self, desirability for other and praiseworthiness values as the original one. However its event parameter is a retrieval event, that in turn has its retrieved event parameter set to the event of the original reaction rule. Take the case of $re1$ and $re2$ whose textual representation is presented bellow. Reaction rule $re2$ would automatically be created if $re1$ was created.

```
re1: [EmotionalReactionRule]
   _event: EventFallHole,
   _desirabilityForSelf: -6,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
re2: [EmotionalReactionRule]
   _event: EventRetrieval - EventFallHole,
   _desirabilityForSelf: -6,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
```

Regarding the matching process, if a reaction rule's event is a retrieval event (retrieval reaction rule), and the event to be matched is not of type "Retrieval", the comparison value is, as described in the generic case, 0 (no match). However, if the event to be matched is of type "Retrieval" a second comparison must take place: the retrieval event parameter of the reaction

rule's event must be compared with the retrieval event parameter of the event to be matched. In this case, similarly to the witness event situation, the result obtained for the retrieval event parameters is used for matching the reaction rule to the original event. For instance, a retrieval event "EventRetrieval - EventFallHole[l(C1: 150,C2: 350)]" would have a positive match with reaction rule $re2$ (presented above).

In the previous chapter we considered that the agent could appraise a past event according to its current appraisal structures (in this case the reaction rules), or reusing the emotional reaction to the event when it first occurred. Although the implementation's description may seem closer to the former perceptive, there is no real distinction in our case. The current implementation is only prepared for a static set of reaction rules defined at the beginning of the simulation. It is in this phase that the additional retrieval reaction rules are created. As the complete set of reaction rules does not change during simulation, reusing the old emotional reaction or using the retrieval reaction rules would not make a difference: the emotional reaction generated by the reaction rules would be the same as the old reaction. Nonetheless, we believe that the implementation can be easily adapted to support a dynamic set of reaction rules, a step that was not taken because it was not envisioned to be relevant in the model's evaluation. With a dynamic set of reaction rules, reusing the old emotional reaction or reevaluating the event according to the current set of reaction rules would give different results.

As a final note, care should be taken in making the reaction rule's set too dynamic. If the reaction rules change constantly, an observer of the agent's behavior will probably have conflicting expectations concerning this behavior. This will ultimately render the task of creating a mental model of the agent's character an impossible task, which will affect the agent's perceived believability [42].

Step 2 - Generating emotions from emotional reactions

Returning to the architecture's description, an emotional reaction is generated when an event and a reaction rule match. Generated emotional reactions are used to create potential emotions. This process starts *step 2* of the Recollective Experience model described in Section 4.3. A potential emotion has the parameters presented in Table 5.4.

An emotional reaction can generate a maximum of three potential emotions. The maximum is three because each emotional reaction can elicit at most one emotion of each of the following three categories of the OCC model (see Figure 2.1 in pg.8): focus on consequences of events for others (HappyFor, Resentment, Gloating and Pity), focus on consequences of events for self when prospects are irrelevant (Joy and Distress) and focus on actions of agents (Pride, Shame, Admiration and Reproach). We will name these three categories *focus on others*, *focus on self* and *focus on actions*, respectively.

If the *desirability for self* of an emotional reaction is not *null appraisal value*, a potential emotion of the *focus on self* category will be generated. If *desirability for self* is negative the potential emotion's type will be Distress, if it is positive the emotion type will be Joy. In both cases the base potential will be the absolute value of *desirability for self* ($base\ potential = |desirability\ for\ self|$). In this category, as well as in the other two, the event parameter is always set to the emotional reaction's event.

Table 5.4: Potential Emotion Parameters

Parameter	Description
event	Event that generated the emotional reaction.
base potential	Scalar between 0 and 10 that represents the potential intensity of the emotion.
type	Enumerate that represents the emotion type according to the OCC model [43]. Possible values are: Joy, Distress, Love, Hate, HappyFor, Resentment, Gloating, Pity, Pride, Shame, Admiration, Reproach, Hope, Fear, Satisfaction, Disappointment, Relief, FearsConfirmed, Gratification, Regret, Gratitude and Anger. However, the implementation only generates potential emotions of the following types: Joy, Distress, HappyFor, Resentment, Gloating, Pity, Pride, Shame, Admiration and Reproach.
valence	POSITIVE if the emotion type is Joy, Love, HappyFor, Gloating, Pride, Admiration, Hope, Satisfaction, Relief, Gratification or Gratitude. NEGATIVE for all other types.

If both *desirability for self* and *desirability for other* of the emotional reaction are different from *null appraisal value*, a potential emotion of the *focus on other* category will be generated. The type of the potential emotion is defined according to the values of *desirability for self* and *desirability for other*:

- *HappyFor*: $desirability\ for\ self > 0$ and $desirability\ for\ other > 0$;
- *Gloating*: $desirability\ for\ self > 0$ and $desirability\ for\ other < 0$;
- *Resentment*: $desirability\ for\ self < 0$ and $desirability\ for\ other > 0$;
- *Pity*: $desirability\ for\ self < 0$ and $desirability\ for\ other < 0$;

In all these cases the base potential is given by the following expression:

$$base\ potential = \frac{|desirability\ for\ self| + |desirability\ for\ other|}{2}$$

Lastly, if the *praiseworthiness* of the emotional reaction is different from *null appraisal value*, a potential emotion of the *focus on actions* category will be generated. The base potential will be the absolute value of *praiseworthiness* ($base\ potential = |praiseworthiness|$). The type of the potential emotion is defined according to the values of *praiseworthiness* and to the emotional reaction's event type:

- *Pride*: $praiseworthiness > 0$ and $event\ type \neq EventWitness$;
- *Admiration*: $praiseworthiness > 0$ and $event\ type = EventWitness$;
- *Shame*: $praiseworthiness < 0$ and $event\ type \neq EventWitness$;
- *Reproach*: $praiseworthiness < 0$ and $event\ type = EventWitness$;

Notice that witness events (events of type *EventWitness*) are treated differently. In witnessed events the agency element of the event is not the agent, therefore potential emotions caused by an emotional reaction to them should be directed outwards (Admiration or Reproach) and

not inwards (Pride or Shame). However, less can be said about events whose type is different from EventWitness. In this case we are assuming that the agent is the agency element of the event, but that might not be so. Take the case of "EventBitBySnake[l(C1: 450,C2: 150)]": it is not a witness event, nonetheless the agent is not the agency element of the event (the snake is). Consequently the implementation does not directly support emotions of type Admiration or Reproach when the agent is directly involved with the event.

Still, this problem can be bypassed by coding this flavor of events as witness events, even if they are not strictly so. In the snake biting situation, at sensor level two events can be generated: the original one ($e1$) and a witness event $ew1$ (see the elements' textual description bellow). Matching a reaction rule $r1$ with $ew1$ would cause an emotional reaction $er1$. The event of emotional reaction $er1$ has type *EventWitness* and the praiseworthiness parameter is negative, consequently a potential emotion pe of type Reproach would be created.

```
e1 : EventBitBySnake[l(C1: 450,C2: 150)]
ew1: EventWitness - EventBitBySnake[l(C1: 450,C2: 150)]
r1 : [EmotionalReactionRule]
       _event: EventWitness - EventBitBySnake,
       _desirabilityForSelf: NULL_APPRAISAL_VALUE,
       _desirabilityForOther: NULL_APPRAISAL_VALUE,
       _praiseworthiness: -5
er1: [EmotionalReaction]
       _event: EventWitness - EventBitBySnake[l(C1: 450,C2: 150)],
       _desirabilityForSelf: NULL_APPRAISAL_VALUE,
       _desirabilityForOther: NULL_APPRAISAL_VALUE,
       _praiseworthiness: -5
pe : [PotentialEmotion]
       _event: EventWitness - EventBitBySnake[l(C1: 450,C2: 150)],
       _potential: 5
       _type: Reproach
       _valence: NEGATIVE
```

After a potential emotion is created, independent of which category it belongs to, its base potential might be recalculated. This happens when the event parameter is a retrieval event. In this case the base potential is recalculated according to the following expression:

$$base\ potential = previous\ base\ potential \times memory\ retrieval\ intensity\ bias$$

In which *memory retrieval intensity bias* is a configurable positive value smaller than one. By using such an expression the base potential of potential emotions generated from emotional reactions to retrieval events, will be smaller in comparison to ones for which the event is a non-retrieval event. Consequently, when an agent reappraises a past event the base potential of the corresponding potential emotion will be smaller than the base potential of the potential emotion originally generated when the past event was appraised. This formula tries to encode the idea, described in the model, that the memory retrieval's experience is, in general, less intense than

the original experience [38].

To illustrate the effect of the base potential's recalculation, consider that event $ew1$, of the previous example, is stored in a memory trace and is afterwards retrieved by location ecphory. Location ecphory generates a retrieval event $ew1r$. One should remember that for each reaction rule concerning a non-retrieval event there is a corresponding reaction rule concerning a retrieval event. Hence, if $r1$ is part of the reaction rule set, then reaction rule $r1r$ should also be there. Reaction rule $r1r$ will have a positive match with event $ew1r$ and as a consequence emotional reaction $er1r$ will be generated. Then, from $er1r$ a potential emotion *per* will be created with base potential equal to 5. Considering that the memory retrieval intensity bias was set to 0.75, the base potential of *per* will be recalculated to 3.75 (0.75×5).

```
ew1r: EventRetrieval - EventWitness - EventBitBySnake[l(C1: 450,C2: 150)]
r1r: [EmotionalReactionRule]
       _event: EventRetrieval - EventWitness - EventBitBySnake,
       _desirabilityForSelf: NULL_APPRAISAL_VALUE,
       _desirabilityForOther: NULL_APPRAISAL_VALUE,
       _praiseworthiness: -5
er1r: [EmotionalReaction]
       _event: EventRetrieval - EventWitness - EventBitBySnake[l(C1: 450,C2: 150)],
       _desirabilityForSelf: NULL_APPRAISAL_VALUE,
       _desirabilityForOther: NULL_APPRAISAL_VALUE,
       _praiseworthiness: -5
per : [PotentialEmotion]
       _event: EventWitness - EventBitBySnake[l(C1: 450,C2: 150)],
       _potential: 5 -> 3.75
       _type: Reproach
       _valence: NEGATIVE
```

For each potential emotion, Appraisal determines a potential according to the base potential and to the current mood. Mood is a scalar value that varies between -10 and 10. Low values represent a bad mood and high values represent a good mood. If the valence of the emotion is positive expression 1 is used, if it is negative expression 2 is used (*mood influence on emotion* is a positive scalar smaller than one).

1. $potential = base\ potential + mood \times mood\ influence\ on\ emotion$
2. $potential = base\ potential - mood \times mood\ influence\ on\ emotion$

The use of the first expression causes positively valanced emotions to be favored by a positive mood and impaired by a negative mood. Conversely, the second expression causes negatively valanced emotions to be favored by a negative mood and impaired by a positive one. For example, consider that the current mood is -5, that mood influence on emotion is 0.3, and that a potential emotion is generated. This potential emotion has a HappyFor type and a base potential of 3. The first expression will be used because the emotion is positively valanced (see Table 5.4). The resulting potential will be 1.5 ($3 - 0.3 \times -5$). If the calculated value would happen be negative, the potential would be automatically set to 0.

After calculating the potential, we check if it is greater than the emotional threshold for the potential emotion's type. A potential emotion is discarded if its calculated potential is smaller than, or equal to, the emotional threshold. Emotional thresholds are used to represent that an agent can have a greater tendency to feel certain types of emotions in comparison with others. For instance, consider that an agent has an emotional threshold of 3 for emotions of type Pride and an emotional threshold of 7 for emotions of type Shame. A potential emotion of type Shame and potential 5 will be discarded, while a potential emotion of type Pride with the same potential will not.

All non-discarded potential emotions are considered regular emotions, having all the parameters in Table 5.4 apart from base potential. Regular emotions have intensities instead. Intensities are calculated using the following expression: $intensity = potential - threshold$. The *threshold* being the emotion type's threshold. Finally, the value range of the intensity ([0;10]) is enforced. If the intensity is greater than 10, it is reset to 10. If it is smaller than 0, it is reset to 0.

Step 3 - Integrating generated emotions into the emotional state

The third step of the Recollective Experience, as described in the model, consisted of integrating the generated emotions into the emotional state (see Section 4.3). In our implementation, the emotional state consists of the already mentioned mood value and of a set of active emotions. These two elements are changed by generated emotions.

Each created emotion is compared to all emotions in the active emotions set. If there is an emotion in the set with the same type and event parameter values as the new emotion, a reinforcing process takes place. If not, the new emotion is simply added to the active emotions set.

The reinforcing process consists of discarding the new emotion and increasing the intensity of the emotion already present in the emotional state. First a new potential is calculated using the emotion's current intensity, the threshold for the emotion's type, and the potential (not the intensity) of the new emotion (*potential'*).

$$potential = \ln(e^{intensity+threshold} + e^{potential'})$$

Then the new potential is used to calculate the emotion's new intensity with the expression previously presented: $intensity = potential - threshold$.

Whether an emotion is simply added to the emotional state, or reinforces an existing emotion, the calculated intensity influences the current mood. For reinforced emotions we consider the intensity after reinforcing. The mood is recalculated according to one of these two expressions.

1. $mood = mood' + intensity \times emotion\ influence\ on\ mood$
2. $mood = mood' - intensity \times emotion\ influence\ on\ mood$

In which *emotion influence on mood* is a positive value smaller than one, and $mood'$ is the mood value before recalculation. If the emotion's valence is positive, the first expression is used, so that positive emotions increase the mood. If the emotion's valence is negative, the second expression is used, so that negative emotions decrease the mood. Additionally, if the new mood

value is greater than 10, it is reset to 10, and if it is smaller than -10, it is reset to -10. In this way we enforce the mood's value range ([-10,10]).

For example, consider that the *emotion influence on mood* is 0.3, the current mood is 8, and that a new emotion of type Admiration and intensity 7 has just been added to the emotional state. The emotion's type is Admiration therefore the valence is positive (see Table 5.4). As a result the mood is recalculated according to expression 1. The value obtained is 10.1 ($8+7\times0.3$), which is greater than 10. Consequently the mood will be reset to 10.

Finally, for each generated emotion, and for each emotion that was reinforced, a memory trace is created, apart from emotions caused by a retrieval event. This exception was put in place to avoid recursive memory retrieval, a problem already mentioned in the model. Created memory traces will have their event set to event parameter of the emotion. The emotion parameter will be set to a copy of the emotion so that when the emotion's intensity changes, it will not change in the memory trace. Lastly, the time stamp will be defined as the current simulation time.

Created memory traces are stored in the general memory traces set, and will initially be in STM (see the Memory Storage section). After they have been passed to LTM they can be selected by location ecphory and the memory trace's event can be appraised a second time.

5.1.6 Emotion and Mood Decay

The processes about to be described are no longer part of the agent's Recollective Experience, however they are part of the agent's emotional experience. Previously, we showed how emotions are added to the active emotions set. We have yet to mention how, and when, they are removed from this set. At each time step, the intensity of all active emotions is decreased. The new intensity is the following:

$$intensity = intensity' \times e^{-emotion\ decay\ factor \times decay \times \Delta t}$$

In which *emotion decay factor* is a positive value smaller than one, $intensity'$ is the emotion's intensity before this recalculation, *decay* is the decay value for the emotion type, and Δt is the time elapsed since the last time step. By attributing different decay values to different emotion types we are representing that certain emotions will fade faster in the agent's emotional state. Ultimately, the decay values enable us to partially define the personality of the agent. For example, for an agent prone to forgive others one may set a high value for the decay of Resentment. In this way the agent feels the Resentment emotions for a short period of time.

When doing the intensity recalculation, if the absolute value falls bellow a *minimum emotion intensity* (positive scalar smaller than one) the emotion is considered to be inactive and is removed from the set of active emotions.

$$Intensity(emotion) < minimum\ emotion\ intensity \iff \text{Emotion is inactive}$$

Along with the active emotions, the mood also suffers a decay effect. At each time step the mood value is recalculated according to the following expression:

$$mood = mood' \times e^{-mood\ decay\ factor \times \Delta t}$$

In which *mood decay factor* is a positive value smaller than one, $mood'$ is the mood's value before recalculation, and Δt is the time elapsed since the last time step. Consequently, recently elicited emotions have a greater effect on mood than older emotions, as proposed by Picard [46].

With the description of the the decay processes in the emotional state, we finish analyzing the modules that are directly related with the model presented in the previous chapter. The only element of the architecture that has not been reviewed was the Behavior module. Its analysis was postponed because it is highly dependent of the application in which the agent architecture was integrated.

5.2 Application

The application started as a course project [44] motivated by the author's thesis. In this project the course's library was used [37]. The project was extended, debugged, improved in reliability and changed to be more agent oriented. The application consisted of a game in which the player controls an avatar (*meemo captain*) that can issue commands to several non-player characters (*meemo minions*). The objective is to lead the meemo minions through each level to an exit point. The avatar and meemo minions should not be hurt in the level.

A map level is constituted of walls, holes, floor, traps, levers, doors and exit points. Maps are loaded from a text file in which each character represents an object tile. In fact a map seen in a top-down view is a rectangle divided in tiles. We will look closer at one type of the elements of a map: traps.

There are two types of traps: rock traps and hole traps. A rock trap consists of a rock that falls from the ceiling when a meemo passes under it. These traps remain undetectable except when they are activated. The rock is pulled to the ceiling after it has hit the meemo, so it can be activated several times. The other type of traps (hole traps), consist of a trap door on the floor. They become visible after being activated and can be avoided afterwards.

Turning to the characters, the meemo captain's behavior is defined by the architecture presented and by player commands. The player is able to control the navigation of the meemo captain and make him issue commands to individual meemo minions. The main command that can be used is calling a meemo. This results in the meemo minion trying to get to the meemo captain's location. The meemo minion's behavior is mainly defined by the architecture described. In addition minions respond to the player's commands via the meemo captain.

Connecting the application with the architecture, we will start by describing how the agents express their emotional state and then continue to define how they choose their paths. This last point is only relevant for meemo minions, as the meemo captain's path is chosen by the player.

5.2.1 Emotion Expression

Although there can be more than one active emotion, only the most intense emotion influences the behavior. This decision was motivated by the idea that emotions should be clearly identifiable on believable agents [4]. Agents express the most intense emotion through a facial expression (see Fig. 5.4). Graphical content for four expressions was created: neutral (that acts as a baseline for the other expressions), sadness, happiness and anger. The choice of the three last expressions

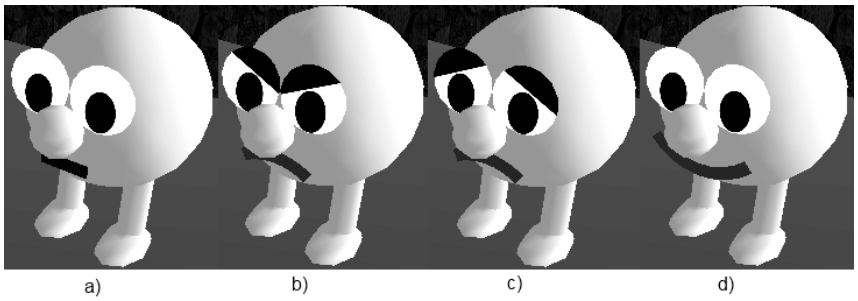


Figure 5.4: Meemo's face expressions: a) neutral b) anger c) sadness (Distress/Pity) d) happiness (Joy/HappyFor)

was motivated by the fact that they are part of the group of six universal facial expressions (anger, disgust, fear, happiness, sadness and surprise) [41][3].

We mapped the emotion types to the available emotion expressions. It should be reminded that each emotion type, taken from the OCC model [43], represents a family of emotions. The emotion type Joy, for instance, represents emotional states such as happy, glad, delighted, pleased, etc. On the other hand, Distress represents emotional states such as sad, unhappy, feeling bad, displeased, dissatisfied, etc. With this in mind, we mapped the emotion type Joy to the *happiness* expression and the emotion type Distress to the *sadness* expression. Consequently when the most intense emotion (emotion with the highest value for the intensity parameter) in the active emotions set is of type JOY, the agent will display a *happiness* expression (Fig. 5.4d). If the most intense emotion in the active emotions set is of type Distress, the *sadness* expression will be displayed (Fig. 5.4c). If there are no emotions in the active emotions set, the *neutral* expression will be shown (Fig. 5.4a). Unfortunately emotions of type Angry are not supported by our implementation, hence *angry* expression is never selected.

Turning to other supported emotion types, HappyFor represents emotional states happy-for, delighted-for, pleased-for, etc. One can notice that these emotional states are quite similar to the ones presented for JOY. Thus, HappyFor was also mapped to the *happiness* expression. Greater care has to be taken with Pity. Pity represents emotional states compassion, sympathy, sad-for, sorry-for, etc. It has been claimed that emotions such as compassion and sympathy have a different facial display pattern than distress [41]. However, this pattern seems to include oblique eyebrows, which are part of the *sadness* expression. Consequently Pity was also mapped to *sadness* expression. Lastly, several emotion types supported by the implementation do not have a mapped facial expression (Resentment, Gloating, Pride, Shame, Admiration and Reproach).

In addition to the facial expression, when the most intense emotion was generated by a retrieval event, a though balloon is presented (see Figure 5.5). An image is displayed on the thought balloon representing the remembered event. In Figure 5.5 the retrieved event was witnessing another agent falling into a trapdoor.

Still concerning the emotional display, although surprise is not directly supported by the implementation, the agents express it nonetheless. When a witness event is generated, the agent does a small hop, turning on its own vertical axis to face the location where the event took place.

[3]Some critique has been set forth to the extent of their universality. However the idea that they have worldwide recognizability, even if with variations due to factors such as culture, still has a strong acceptance.

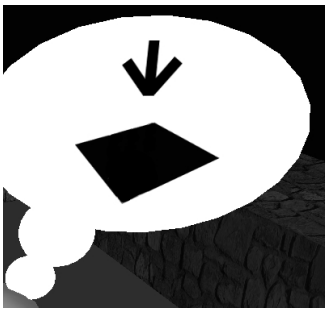


Figure 5.5: Meemo's thought balloon

Furthermore, if the agent is following a path and a witness event is generated by its sensors, the following happens: the agent does a small hop turning to face the event's location; waits for *meemo wait duration* seconds (configurable value); and then restarts walking to the destination it was heading before. However, in this situation the path is recalculated because the witnessed event might have caused an emotion, that in turn would be stored in memory together with the event information. This memory might affect the agent's path choice (see next section).

Lastly, the agent's mood is also graphically expressed: it defines the agent's color saturation. The lower the saturation of a color, the closer it will be to a gray tone. The word "gray" can be used to classify a mood [27] describing it as negative. In fact the gray tones are sometimes associated with a negative state of mind. Due to this associated meaning, we decided that the lower the agent's mood was, the lower its color saturation should be. Taking that the saturation varies between 0 and 1, and the mood between -10 and 10, the saturation percentage is calculated by the following expression:

$$saturation\ percentage = \frac{mood + 10}{2 \times 10}$$

5.2.2 Path Planning

Besides considering the agent's emotional state, the Behavior module also takes into account memories when the agent is choosing a path. We use an A* pathfinding approach in which the node cost is influenced by stored memories (for a detailed description of A* consult [50]). The pathfinding algorithm uses a discretization of the virtual world as a planar grid (See Figure 5.6). Each node of the search graph corresponds to a position of the grid. We consider that a position has a maximum of four adjacent positions (A is adjacent to B, C, D and E). Consequently, when expanding a node a maximum of four child nodes will be generated. We do not consider eight adjacencies to avoid paths that pass through possibly occupied corners. For example, consider that we are expanding a node corresponding to position D, and that position F is occupied by an obstacle that completely fills the grid square. If we were to consider the adjacency between D and B, we would risk that the agent, when using the path, would get stuck in the corner of F's obstacle. As can be verified in the figure, certain positions have less than four adjacencies due to obstacles and to the virtual world borders. For instance, position D only has three adjacencies, as well as G, and H only has two adjacencies.

During search, the A* algorithm needs to know the cost of getting from one position to an adjacent one. For the purpose of calculating such cost, we define an auxiliary cost of crossing a

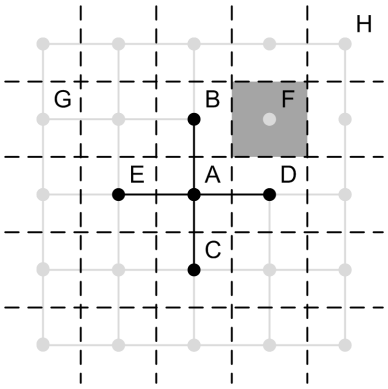


Figure 5.6: Discretization of the game world as a planar grid. Dots represent positions, edges represent connections between positions and dashed squares indicate the space that can be occupied by an entity in a certain position.

grid square from one edge to the opposite one. We assume that crossing the square horizontally or vertically has the same cost. This value is associated with the position inside the corresponding square. The cost of moving between two adjacent positions is calculated as half of the cost associated with one of the positions plus half of the cost associated with the other position. Take the case in which A has associated cost 30 and D has associated cost 40. The cost of getting from A to D would be 35 ($\frac{30}{2} + \frac{40}{2}$).

Before the search starts, the cost of all positions not occupied by an obstacle is set to *move cost neutral* (positive parameterizable in a configuration file). Then, for each stored memory trace, we calculate which grid position is closer to the memory trace's event location. If the memory trace's emotion is negatively valanced, the position cost is updated according to the following expressions:

$$move\ cost\ change = (move\ cost\ maximum - move\ cost\ neutral) \times \frac{Intensity(Emotion)}{10}$$

$$move\ cost = min\{move\ cost' + move\ cost\ change, move\ cost\ maximum\}$$

In which *move cost'* is the old cost associated with the position, and *move cost maximum* is the maximum cost of crossing a square (positive parameterizable in a configuration file). As a result when the agent has a negative memory (memory trace with a negatively valanced emotion parameter) associated with a certain location (location of the event parameter) it will try to avoid that location when choosing a path. In addition, the more intense the memory trace's emotion was, the harder it will try to avoid the location.

If on the other hand, the memory trace's emotion is positively valanced, the position cost is updated according to the following expressions:

$$move\ cost\ change = (move\ cost\ neutral - move\ cost\ minimum) \times \frac{Intensity(Emotion)}{10}$$

$$move\ cost = max\{move\ cost' - move\ cost\ change, move\ cost\ minimum\}$$

In which *move cost minimum* is the minimum cost of crossing a square (positive parameterizable in a configuration file). Consequently, when choosing a path the agent will favor those that pass

by locations with an associated positive memory (memory trace with a positively valanced emotion parameter). This bias will be proportional to the intensity of the memory trace's emotion. Finally, stored memory traces can have close by event location parameters, therefore a grid position cost may be influenced by several memory traces.

Besides determining the cost of moving between adjacent positions, A* also needs an heuristic to estimate the cost of reaching a position from another position. The heuristic used is the euclidean distance between positions times the *move cost minimum*. When calculating the euclidean distance the length of the square grid edge is used as a unit. For instance, considering that the *move cost minimum* was set to 2, the cost estimate of reaching H from A will be approximately 8.49 ($2 \times \sqrt{3^2 + 3^2}$).

5.3 Complete Scenario

5.3.1 Introduction

Having described all the elements of the agent architecture, and having given an introduction to the application, we can now describe a complete scenario of the whole working system. Consider a game scenario in which the level exit is closed, and to open it a lever must be pulled by two meemos. The level map is constituted of rooms, corridors, an exit, a lever, and a trap (Figure 5.7).

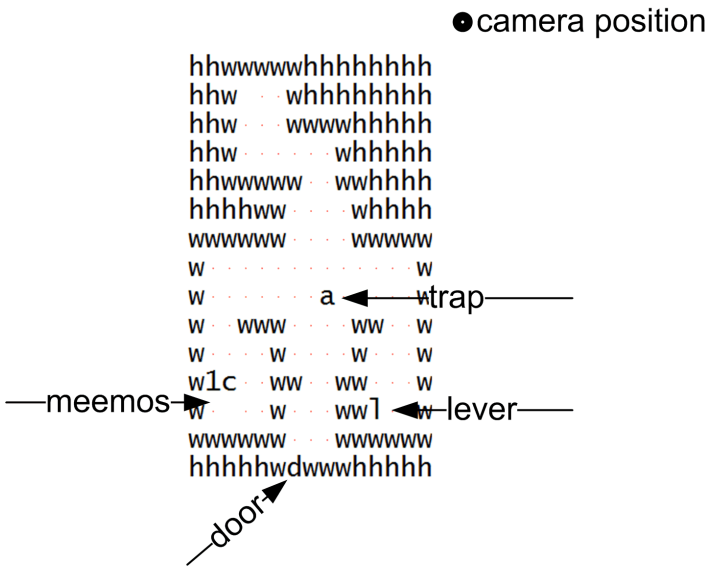


Figure 5.7: Complete Scenario Map (top-down view) - The correspondence between characters and objects is the following: "w" is a wall, "h" is an inaccessible space, "a" is a trap, "l" is a lever, "1" is the minion, and "c" is the captain.

Two meemos are in the scenario: one minion and one captain (the player's avatar). Besides controlling the captain, the player is also able to control an overview camera. Consequently he can see where the exit is and where the lever is. However he can not see where the trap is unless it is activated. To simplify the description, the avatar's actions that are the consequence of commands issued by the player will sometimes be referred to as if the player was himself performing them. For example, instead of saying "the player moved the meemo captain to

location X" we would say "the player moved to location X".

Both meemos have the following three reaction rules concerning non-retrieval events:

```
rr1 : [EmotionalReactionRule]
       _event: EventHitByRock,
       _desirabilityForSelf: -4,
       _desirabilityForOther: NULL_APPRAISAL_VALUE,
       _praiseworthiness: NULL_APPRAISAL_VALUE
rr2 : [EmotionalReactionRule]
       _event: EventWitness - EventHitByRock,
       _desirabilityForSelf: -1,
       _desirabilityForOther: -4,
       _praiseworthiness: NULL_APPRAISAL_VALUE
rr3: [EmotionalReactionRule]
   _event: EventReachExit,
   _desirabilityForSelf: 5,
   _desirabilityForOther: NULL_APPRAISAL_VALUE,
   _praiseworthiness: NULL_APPRAISAL_VALUE
```

They also have three additional reaction rules concerning retrieval events that are the counterparts of the just presented rules. We will name them *rr1r*, *rr2r* and *rr3r* respectively. Additionally, both meemos start with mood set to zero and with no emotion in the active emotions set. Consequently they start with a saturation percentage of 0.5 and a neutral expression. Finally, both have the following emotional thresholds: 1 for emotion type Distress, 2 for emotion type Pity, and 1 for emotion type Joy.

5.3.2 Trap Activation

The two meemos are in a room (Figure 5.8). The player goes out of that room to where there is a trap. He then calls the meemo minion. Next the player inadvertently moves to a spot where there is a trap. This causes the trap to be triggered and the meemo captain to be hit by the rock.

Diving into the agent's mental processes, the meemo captain's sensors generate an event of type "HitByRock". This event will have a positive match with reaction rule *rr1* generating a potential emotion of type Distress with base potential 4 ($|-4|$). The calculated potential will also be 4 because the current mood value is 0. Since the the potential is higher than the threshold for Distress (2), the intensity will be set to 2 ($4-2$). As there is no active emotion, this new one will simply be added to the active emotions set. Then the mood is decreased because the emotion's valence is negative. Additionally, a memory trace with the event information and a copy of the emotion is stored in general memory traces. Abstractly, it will remain in STM for the duration of *short term duration*.

Concerning meemo captain's Behavior, as the only emotion felt by it is of type Distress, its expression will change from *neutral* to *sadness*. Still in Behavior, the mood decrease the color saturation of the character to also decrease.

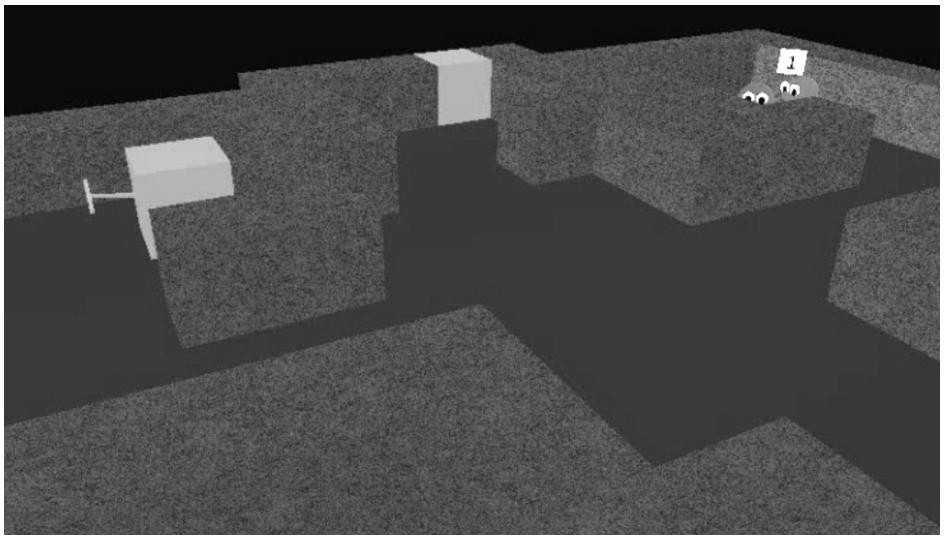


Figure 5.8: Complete Scenario I - Beginning.

Meanwhile, the meemo minion witnessed meemo captain being hit, consequently it does a small hop turning to meemo captain's location. The event generated by the sensors will match reaction rule *rr2*, creating two potential emotions: one of type Distress and one of type Pity. The first emotion will have base potential 1 ($|-1|$). The potential will also be 1 because the agent's mood is zero. One is not greater than the threshold for Distress (1) hence the potential emotion is discarded.

On the other hand, the Pity emotion will have psotential 2.5 ($|1|+|4|$) which is greater than the threshold for Pity (2). Thus an emotion of type Pity and intensity 0.5 ($2.5-2$) will added to the active emotions set, and the meemo minion's mood will decrease (Pity has a negative valence). This emotional change will result in a decrease of the agent's color saturation and the character's expression will change to *sadness*, as the most intense emotion is Pity (see Figure 5.9). Moreover, the event together with a copy of the caused emotion are stored as a memory trace in Memory Storage.

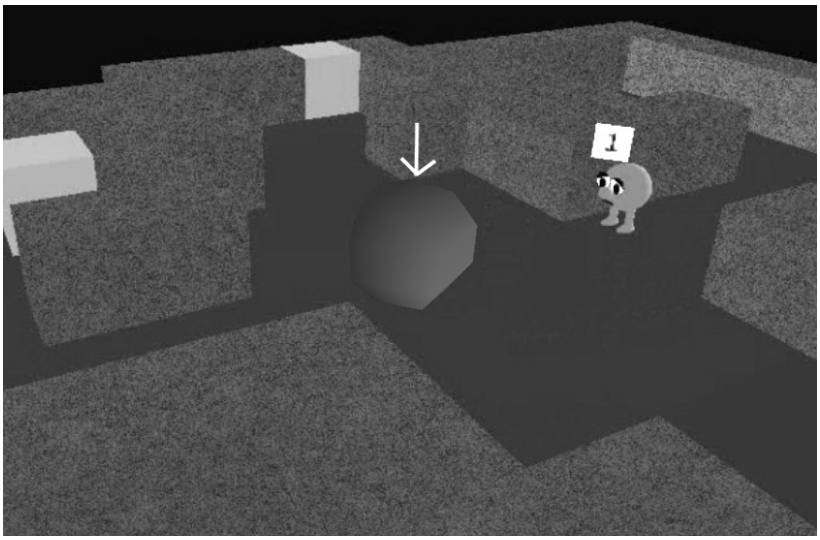


Figure 5.9: Complete Scenario II - Captain caught in trap.

Returning to the player's interaction, seeing the meemo captain being hit by the trap, the player moves it away from the spot. Then he calls the meemo minion so it will come closer to the meemo captain. If the meemo minion was to move in a straight line in direction of the meemo captain, it would pass under the trap, activating it a second time. However, when planning its path, in its grid division the trap position has a high cost associated to it. This high cost is due to the memory trace concerning the emotion Pity and the event of witnessing meemo captain being hit. Consequently, the path chosen goes around the trap spot, avoiding triggering the trap.

Remember that both meemos have a memory trace each in memory storage. Furthermore, the location of the event parameter of these memory traces is the trap location. Meemo Captain might still be close to the location, and meemo minion has just passed by it. Thus, in both agents there is an ecphoric match. Nonetheless, considering that the time passed since the trap activation is smaller than *short term duration*, the memory traces are still in STM. Consequently no memory retrieval occurs.

5.3.3 Going to the Lever

Turning again to the player, he moves the meemo captain to the lever and calls the meemo minion. The meemo minion plans a path to the captain and then follows this path. Following this, the player commands the meemo captain to pull the lever together with the meemo minion. The two meemos pull the lever and as a consequence the door to the exit opens (see Figure 5.10).

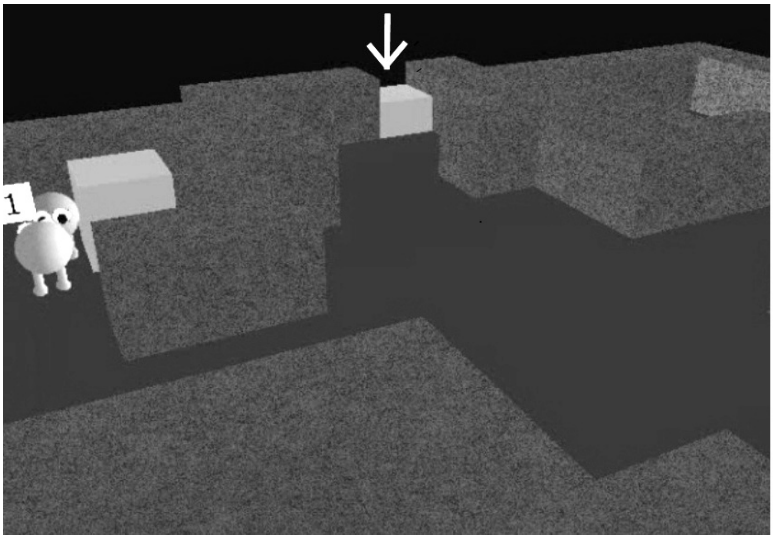


Figure 5.10: Complete Scenario III - Exit opening.

Meanwhile, consider that the emotion intensity of meemo minion's Pity emotion and meemo captain's Distress emotion have decayed to values smaller than *minimum emotion intensity*. As a result they have been removed and both characters now display a *neutral* expression. Additionally consider that the mood of both meemos has decayed to zero.

5.3.4 Trap Spot Revisited

At this time point, the player moves towards the exit and in doing so passes close by the trap spot. There is an ecphoric match of the memo captain's memory trace with the agent's current location. Suppose that *short term memory duration* seconds have already passed since the trap's activation. This means that the memory trace is in LTM and can be retrieved. Accordingly a retrieval event is generated by Location Ecphory which is fed to Appraisal.

The event then has a positive match with rr1r resulting in the creation of a potential emotion of type Distress. Considering that the *memory retrieval intensity bias* is 0.75, the base potential is set to 3 (4×0.75) because the reaction was to a retrieval event. As currently the mood is 0, the potential will be equal to the base potential (3). This base potential is greater than the threshold for Distress (2) consequently an emotion with intensity 1 ($3 - 2$) is generated and added to the active emotions set. The emotional state also changes in what concerns the mood: it decreases because the emotion is negatively valenced. These changes lead the meemo captain to display a distress expression and have its color decrease in saturation.

Due to the fact that the event causing the emotion was a retrieval event, no new memory trace is stored at this time point. However, due to the same fact, the character displays a thought balloon above its head with the representation of a rock falling (see Figure 5.11).

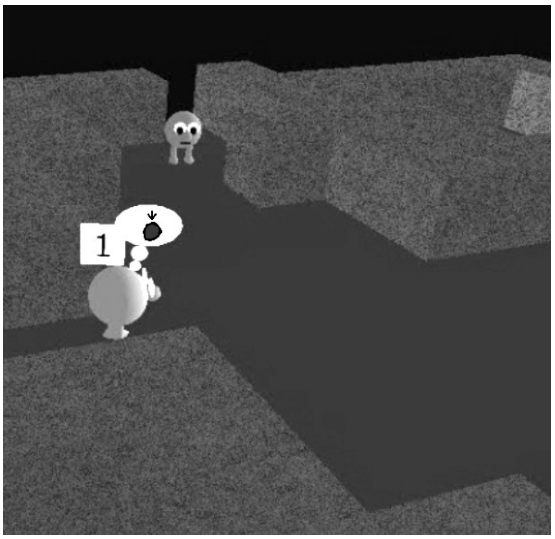


Figure 5.11: Complete Scenario IV - Trap Spot Revisited

Returning to the player's interaction, he calls the meemo minion still standing by the lever. This meemo also passes close by the trap location causing a location ecphory. The resulting retrieval event will match reaction rule rr2r. Consequently a potential emotion of type Pity and one of type Distress are generated. The distress emotion will be discarded as before since the initial base potential is quite small (1). The Pity emotion will have base potential 2.5 that will be recalculated to 1.875 (2.5×0.75). As 1.875 is no longer greater than the threshold (2), this emotion is also discarded.

5.3.5 Reaching the Exit

Then, the player moves the meemo captain to the exit. In doing so a the agent's sensors will generate an event of type "EventReachExit". This event matches reaction rule rr3 causing a potential emotion of type Joy to be generated with base potential 5. The potential is calculated using the current mood. Suppose that currently it is -0.25 and the *mood influence on emotion* was configured to 0.3. The resulting potential is 4.925 $(5 + (-0.25) \times 0.3)$ which is higher than the threshold for Joy (1). Thus a emotion of type Joy and intensity 3.925 $(4.925 - 1)$ is added to the active emotions set. In addition the mood is increased and a memory trace is stored.

The emotion added is more intense than the Distress emotion already in the active emotions set: the distress emotion had initial intensity 1 and its current intensity is even smaller due to decay. Hence the face expression is changed from "sadness" to "happiness". Moreover the thought balloon disappears and the character's color saturation increases.

Finally, the player calls the meemo minion to the exit. The meemo minion moves towards the exit and sees it. A similar process as the just described for meemo captain occurs. As a result meemo minion displays a "happiness" expression and its color increases in saturation. Both meemos have reached the exit so the level ends.

5.4 Concluding Remarks

The purpose of this chapter was to describe how we implemented our conceptual model and how we integrated it in a application that could support the model's evaluation. In the model's implementation we indicated how using two memory traces sets enabled us to simulate the division between LTM and STM. We described the implementation of the location ecphory approach using the euclidean distance and physical locations. Then we presented a modified version of FAtiMA's reactive appraisal that has reaction rules for retrieval events. In addition, it decreases the base potential of emotions caused by retrieval events.

We proceeded by explaining the game application in which the architecture was integrated. Our architecture was used to model the game characters' behavior. This behavior included: expressing facial expressions corresponding to the most intense emotion felt; color saturation variation based on mood; presentation of a thought balloon when the agent's displayed emotion was caused by a retrieval event; path choice avoiding locations where negative events have occurred and favoring paths where positive events have occurred.

Finally we presented an extensive application scenario in which the majority of the elements described in this chapter were applied. A similar scenario was used when preparing the evaluation. Moreover, we will describe in the next chapter how we used the game application to create our test scenario.

Chapter 6

Evaluation

In this chapter we will describe how we tested our hypothesis concerning the effect of episodic memory retrieval on agent believability. Firstly we characterize the preliminary tests performed, in which we accessed the perceptions of a small group of individuals towards agents modeled in our architecture. The feedback from these tests was used in the final evaluation we present subsequently. We describe the participant's group, the evaluation's overall structure and the manipulation done between test conditions. Then we proceed by explaining how we measured believability and present the results obtained. Last, we state some concluding remarks that include the experiment's main results.

6.1 Preliminary Tests

6.1.1 Motivation

As stated in the Introduction, we wanted to analyze the *influence of episodic memory retrieval on agents' perceived believability.* In the application developed, a user does not have direct access to an agent's internal mental process, for the exception of seeing the thought balloon concerning emotions caused by retrieval events. Consequently, his, or her, perception depends mainly on the externalization of this mental process, namely the activation of actuators by the Behavior module. Furthermore, we modeled the retrieval of emotionally relevant events. Consequently, we decided to focus on *individual's identification of agents' emotions, and their interpretation of agents' mental process through behavior.*

6.1.2 Methodology

To get a first perspective into these two aspects we performed preliminary tests. These tests consisted of: watching recorded videos of the application running, reading some information concerning them, answering a questionnaire, verbally answering some informal questions about the experiment and giving open feedback about the application. In fact the test structure was changed iteratively based on feedback, with a total of six versions being made. Seven people (ages between 20 and 60) participated in the preliminary tests with some individuals being exposed to more than one version.

Although the videos' scenarios were different between versions, they were all variations of the complete scenario described in the previous chapter (see Section 5.3). We recorded each scenario with two versions of the application: one in which the memory part of the agent architecture was enabled and one in which it was disabled. Typically the scenario consisted of three meemos trying to escape a dungeon: minion 1, minion 2 and captain. Captain was presented as an agent, although in fact its actions were manually defined during the preparation.

In the scenarios, captain starts by exploring a bit the dungeon and finds the lever that will open the dungeon's exit. Then captain calls minion 1. When following the captain's voice, minion 1 inadvertently sets off a trap, gets hurt, and becomes sad. Nonetheless it is able to continue its path and reach the captain. Meanwhile, minion 2 witnesses this event and becomes sad for minion 2. At this time point captain calls minion 2. Here there are two alternatives: either minion 2 does not have episodic memory, and will also fall in the trap; or it does have it and consequently avoids the trap. Either way, afterwards minion 2 continues its route and reaches the lever. Following this, the three meemos pull the lever simultaneously and as a consequence the exit opens. As some time has passed since the trap activation, the meemos' emotions have decayed and been removed from their emotional state. Then the captain goes to the exit and calls the other two meemos. The path they take does not pass exactly were the trap is, but close by. Here there are again two possibilities: either they do not remember the past events and their emotional state remains unchanged (without episodic memory); or they remember the past events and become sad again (with episodic memory). In both cases the video ends when all meemos have reached the exit.

After recording the videos we divided each one in two parts. The original videos had on average a total duration of about two minutes and each part would had about half of the total duration. Besides the videos, participants were also presented with some text that hints about the implicit goals behind the agent's reaction rules: meemos were presented as being concerned with one another; it was claimed that meemos in the scenario wanted to reach the exit.

6.1.3 Feedback

Although no statistical tests were performed with the preliminary data gathered, much interesting feedback was collected. A crucial feedback point, was that participants were having difficulty in identifying the agents' facial expressions. This problem persisted even when participants were instructed, before the videos, to pay close attention to these expressions. Nonetheless, when presented with examples of meemos' facial expressions participants were typically comfortable with the labels attributed (similar figures to Figure 5.4 were used). Still, when asked which expression an agent expressed in a certain situation, many times participants had difficulty in answering. We believe that there were two causes for this problem: the scenario sometimes occluded the meemos' heads; participants forgot general details concerning the videos. Which leads us to our second feedback point.

In general, it was hard for participants to remember the conducting story of the scenario. When asked to describe the videos, participants' descriptions would sometimes be missing details. In other cases the actions would be described in a wrong order. There was even a case in which a participant claimed to see an event that did not happen. We tried to tackle this problem

by providing additional information in the textual introduction, preparing the participants for the videos' events. However, participants complained that the introduction was getting too large so we returned to the previous format.

Typically each video was seen only once, which might contribute to the above mentioned problem. Nonetheless, had participants been allowed to see each video more times, the experiment duration would probably extend considerably. In fact, reading the textual introduction, watching the videos and completing the questionnaire, took on average about ten minutes. Participants reported being comfortable with this duration and advised not to extend it.

Summing up the feedback, participants: had difficulty seeing the agent's facial expressions; would not remember the scenario's story accurately; and found the questionnaire duration adequate. These three points were considered in the final evaluation.

6.2 Final Evaluation

Our motivation, stemming from video game examples, led us to develop a game application in which we could analyze the effect of episodic memory retrieval on agent believability. Nevertheless, our final evaluation did not contain interaction between the user and the application. This decision was motivated by time-line and resource constraints. First of all, an interactive evaluation would undoubtedly require more time to be performed. Secondly, we wished with this evaluation to reach a large number of participants to have a better change of achieving statistically significant results. We estimated that the potential number of participants in a interactive experiment with the resources available, would probably be less than in a non-interactive on-line experiment. We realized that the results gathered at this time point could later on be used to focus the design of an interactive experience. Lastly, the user interface for the game in the current implementation is rather cumbersome. In an interactive experiment this factor would demand a period of training for users, which would extend the experiment's duration.

Additionally, the influence of memories on path planning was not tested in the evaluation. We chose to do so because the use of memories to avoid places where dangerous events have happen, included on the preliminary tests' scenarios, can be argued as not being episodic memory retrieval. First of all, events are not being re-experienced, as in Tulving's definition of episodic memory retrieval [53]. Secondly the memory information is being used more in a semantic memory perspective: agents' knowledge about the world might include possibly dangerous locations. Hence, as we wanted to evaluate the influence of episodic memory retrieval on agent believability, and not of semantic memory retrieval, the influence of memories on path planning was not demonstrated.

Having justified these two experimental decisions, we now clarify this experiment's objective. Then we give an overview of the experiment's structure and explain how it was performed. Following this, we indicate how we manipulated the independent variable of the experiment between test conditions. Subsequently, we describe the character features that were used to indirectly quantify believability. Finally, we present some results based on the retrieved data.

6.2.1 Objective

With this experiment we wished to get some insight into our main thesis hypothesis:

> Autonomous agents modeled by an architecture that incorporates episodic memory retrieval of emotionally relevant events, will be perceived as more believable, than agents modeled by a similar architecture that does not incorporate episodic memory retrieval.

As mentioned before, we have developed an architecture for autonomous agents that supports episodic memory retrieval. Memories are selected by a location ecphory mechanism and are afterwards re-experienced by a modified appraisal system. This process affects the agents' emotional state that in turn is reflected in their behavior. We developed a game application that uses the mentioned architecture to model the behavior of virtual characters. Our intuition is that the characters' behavior will be perceived as more believable by observers if the retrieval process is active.

6.2.2 Procedure

The experiment was performed on-line with individuals receiving requests for participation by email and through facebook[1] event invitations. There were 96 participants, 95% having ages between 14 and 48 and the remaining having 49 or more (1% non-responses)[2]. Thus the study left out infants and seniors. Moreover, the gender distribution of participants was fairly balanced: 51% male and 49% female (6% non-responses). Furthermore, the participant's country mode was Portugal (58%), with Germany (9%) and The Netherlands (6%) being the second and third more represented countries respectively (3% non-responses). Consequently, the majority of participants were European. Finally, none of the participants of the preliminary tests took part in this final evaluation.

During the experiment, participants were told a simple story involving virtual characters called meemos. This story was transmitted through text, images and videos. The images and videos were captured from our game application, with the characters' behavior being driven by our agent architecture. We decreased the complexity of the presented scenario, comparatively to the ones in the preliminary tests, because of the received feedback. Besides being told the story, participants were also asked to classify the characters' behavior and to identify which facial expressions the characters' displayed. The experiment's structure is now further detailed.

6.2.3 Structure

The experiment can abstractly be divided in seven sections as presented in Figure 6.1. In the Introduction participants were exposed to text, images and a very short video concerning meemos. Essentially they were explained how a hole trap works (see the description in Section 5.2) and that two meemos were about to go through a corridor that had a trap yet to be activated. Following this, participants classified their expectations towards the meemos' behavior ("Expressions Questions").

[1] ©Facebook

[2] Percentages will be presented in regard to valid answers except for percentages of non-responses. Percentages of non-responses are in regard to the total number of participants.

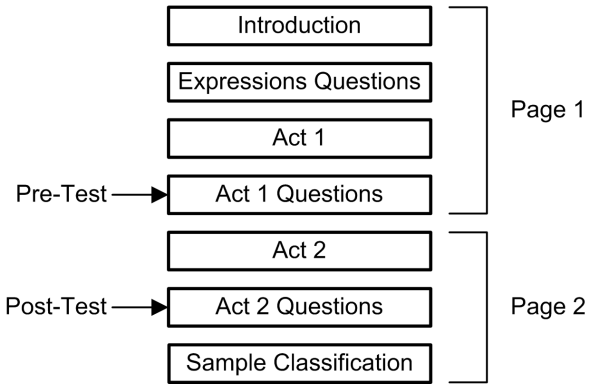


Figure 6.1: Final Experiment Sections

Then participants were instructed to see a video ("Act 1"). In this video two agents are shown walking in a tunnel (meemo 1 and meemo 2) initially with neutral expressions. One of them (meemo 2) falls in a trap and dies, with the other one reacting by showing a *sadness* expression. This expression was selected by the Behavior module of the agent when an emotion of type Pity was added to the emotional state. In turn, this emotion was caused by a reaction rule with negative values of *desirability-for-self* and *desirability-for-other* that matched the event sensed. After seeing the video, participants were asked to classify their perception of meemos' behavior in general ("Act 1 Questions"). In addition they were asked to identify the emotional expression of the characters (before and after the trap being activated).

The short story continued to be told in "Act 2". The participant was explained that some time had passed since the event described in "Act 1". Then he, or she, was instructed to see a video that shows meemo 1 going by the same tunnel and passing close by the trap (now a hole). In this video the character initially has a *neutral* expression. After meemo 1 had passed by the trap, there were three possible endings to the story, corresponding to three test conditions that participants were randomly assigned to:

test condition *retrieval*

The behavior of meemo 1 is driven by our agent architecture. Meemo 1 reacts emotionally when passing close by the trap, displaying a *sadness* expression.

test condition *no retrieval*

The behavior of meemo 1 is simulated as if it was driven by an architecture with reactive appraisal but without episodic memory retrieval. Consequently, meemo 1 does not have any emotional reaction when passing close by the trap.

test condition *random expression*

The behavior of meemo 1 is simulated as if it was driven by an agent architecture with reactive appraisal, without episodic memory retrieval, but with random expression of emotions. Meemo 2 displays a *happiness* facial expression after passing close by the trap. Details on why we chose to simulate the architecture in this way are presented in the next subsection.

Following "Act 2", participants were asked similar questions as the presented in "Act 1 Questions" and three additional ones of the same sort ("Act 2 Questions"). Moreover, they

were asked if they expected an emotional reaction from meemo 1 when passing by the trap, which was this reaction, and to explain why they thought meemo 1 reacted like it actually did. Last, the questionnaire ended with a set of questions concerning the participant ("Sample Classification"). For the the full experiment questionnaire consult Appendix A. We proceed by explaining how we created the different text condition endings.

6.2.4 Test Conditions - detailed

Test Condition *Retrieval*

First we will consider what happened in our architecture, when the system was running for "Act 1". An emotion of type Pity was added to the emotional state, thus a memory trace was added to the general memory traces set. This trace had as parameters the emotion mentioned and a witness event of an agent falling in a hole.

In "Act 2" we state that "some time has passed". To simulate this we led the meemo away from the trap for more than the configured *short term memory duration*. Additionally, we waited enough time for the meemo's emotion to decay and be removed from the emotional state. When the agent passed close by the trap's location the second time, there was an ecphoric match of the stored memory trace. As this trace was on LTM, a retrieval event was generated and fed to the Recollective Experience part of the architecture.

This retrieval event matched with a retrieval reaction rule having negative values for *desirability-for-self* and *desirability-for-other* (this rule had been automatically created from the reaction rule that caused the original Pity emotion). Following this, a new Pity emotion was generated. Consequently the character displayed a *sadness* expression. To create the video for "Act 2" of the test condition *retrieval*, we recorded the system running starting when meemo 1 was getting close to the trap, but before the ecphoric match, and ending a bit after the emotional reaction.

Test Condition *No Retrieval*

To probe our hypothesis, we needed a baseline to which to compare our system. To create such baseline we reenacted "Act 1" and "Act 2", but this time removing the reaction rule concerning witnessing another agent falling in a hole. This caused the agent not to react emotionally to the event in "Act 1", thus not storing a memory trace. Since no memory trace was stored, in "Act 2" the agent did not have an emotional reaction, remaining with a *neutral* expression. We recorded the system running for the "Act 2" part of the story, and used it in the experiment. However, in "Act 1" we reused the original video in which the agent reacts emotionally to the event.

This test condition, that we named *no retrieval*, simulates an architecture in which in spite of agents having emotions, they do not have episodic memory retrieval of emotionally relevant events. Nonetheless, it could be argued that different results for the test conditions *retrieval* and *no retrieval* were caused by the fact that in *no retrieval* the agent expresses emotions less times.

Test Condition *Random Expression*

Motivated by this last argument, one can imagine an alternative architecture that could be used for comparison, in which agents would react emotionally to events, and in addition would express emotions in a randomized sequence. The frequency of such random expression would have to be similar to the frequency that an agent having episodic memory retrieval would typically express an emotion due to retrieval.

An agent whose behavior was driven by such an architecture, compared to an agent whose behavior was driven by our architecture, would: not express emotions when ours would (I); express emotions in certain situations for no apparent reason (II); express emotions when our agent would, but of a different type (III); and in very rare cases express the same emotion as our agent (IV).

Our third test condition, *random expression*, was a simulation of such an architecture. We choose to simulate situation III for "Act 2" because: simulating situation I would be equal to the *no retrieval* video; simulating situation IV would be equal to test condition *retrieval* video; and situation II was harder to simulate given the chosen scenario. To simulate situation III we changed the reaction rule that was causing the Pity emotions: we turned the *desirability-for-self* and *desirability-for-other* to positive values. Then we reenacted "Act 1" and "Act 2" recording the system running during the "Act 2" part of the story. In "Act 1" and "Act 2" an emotion of type HappyFor was generated, causing the agent to display a *happiness* expression. We used the recorded video for the "Act 2" part of the experiment, leaving the original video for "Act 1".

It can be noted, that besides *happiness*, the architecture described could have generated other expressions. We chose *happiness* because we expected it to create a clearer difference in participants perceptions when compared to the test condition *retrieval*. Additionally, the display of thought balloons was disabled in all test conditions, because this was yet another factor that could variate in test condition *random expression*. Ultimately, testing this *random expression* architecture would require participants being exposed to several randomly generated emotion situations, which was impracticable in this experiment's context.

6.2.5 Measures

Believability Features

To compare the described test conditions in terms of their agents' believability, we used what we will call *believability features*. Believability features are the participants' perception of elements that are potential enhancers, or even requirements, for believability. These elements were extracted from believability definitions reviewed in the Background (Section 2.3).

The believability features considered were the following:

- **behavior coherence** - in Ortony's definition [42] coherence is a crucial element for believability. As participants will see the agents' behavior, and not explicitly their internal state, they were asked about the coherence of the first.

- **change with experience** - the agent's change is an element present in Rollings and

Adams' definition [47](pp. 134–135) as well as in Loyall's [33] one. Participants were asked if they perceived this change.

- **awareness** - can be mapped to Lester and Stone's situated liveliness [32] as well Loyall's [33] reactive and responsive elements.

- **behavior understandability** - in Ortony's definition [42] of believability, it is implicit that participants must be able to create a model of an agent's behavior motivations. Furthermore, Bates [4] points out that an agent's actions must express what it is thinking about and its emotional state. For situations in which the thought process is not explicitly shown (e.g. with a thought balloon) this last sentence can be translated to: the agent's actions must express what the participant thinks the character is thinking about. But for this to happen, the spectator must be able to create a model of the character's thought process. Hence, the participant must understand the character's behavior.

- **personality** - the notion of personality is transversal to almost all believability definitions presented. Following Loyall's [33] definition of believability, participants should be able to identify the agent's behavior details that define it as an individual, that make it unique.

- **visual impact** - is the amount by which an agent draws our attention, and was proposed by Lester and Stone as an enhancer of believability [32].

- **predictability** - Lester and Stone also point out the importance of behavior patterns not being recognizable, specially in the context of long term interactions. Moreover, when considering variability, Ortony [42] warns for the harmful effect predictability can have on believability. However, Ortony also stated that complete lack of predictability may affect behavior coherence, and consequently believability. In the end, extreme predictability and extreme unpredictability harm believability.

The use of these features was motivated by the fact that, although many computer science researchers have proposed definitions for believability, few objective evaluations of agent believability have been done. Even Loyall [33] states that evaluating believability is fundamentally subjective (p.167). Nonetheless, he evaluated in a informal way the architecture Hap by the time people remained engaged with a group of agents built in it. In our case, such a metric would be hard to apply, as we decided to pursue a non-interactive experience.

To turn the evaluation of believability into a more objective task, we resorted to believability features. In this way participants were asked about more objective aspects of the agent, instead of believability in general. *It is our belief that increased perception of these features translates into a greater sense of believability, for the exception of predictability.* If participants perceive an agent as highly predictable, or highly unpredictable, this affects negatively its believability.

One can easily realize that not all the believability features mentioned in the Background were considered. Our choice had three main motivations. First of all there is some overlap between definitions, with similar concepts being described in different ways. Secondly, for certain features the agents' behavior is equal for the three conditions. For instance, the timing of the emotion expression is the same for all test conditions. Finally, the use of a larger set of features would

extend the duration of the on-line questionnaire, possibly increasing the number of participants dropping out of the experiment.

Video Game Features

In addition to the believability features, since our motivation was linked with video-game characters, we also analyzed the difference between test conditions in what concerned four features that have been used to characterize this type of characters. These features, that we will name *video game features*, are the following:

- **friendliness** - if the agent is perceived as friendly or hostile. This characteristic is heavily linked with Isbister's notion of agreeableness for video game characters (pp. 24–30) [28].
- **intelligence** - the sense of intelligence the participant gets from observing an agent. The selection of this feature was motivated by the fact that in the preliminary tests participants reported their perception of the agent's intelligence. They would say that agents were "smart" or "dumb", and would compare versions by saying they were "smarter" or "dumber".
- **likability** - the extent to which the participant liked the agent. This feature is referred in Isbister's character evaluation checklist (pp. 274–275) [28]. Likability should not be confused with friendliness. An antihero, defined as "a central character in a novel, play, etc., who lacks the traditional heroic virtues"[22], might be unfriendly but still be liked.
- **interest** - by the participant in the character. This feature is also present in Isbister's character evaluation checklist as well as in Rollings and Adams' requirements for a main character [47](pp. 134–135).

Feature Phrases

The participants' perception of the believability features and of the video game features were evaluated comparing answers from two parts of the experiment: "Act 1 Questions" that we will refer to as pre-test; and "Act 2 Questions" that we will name post-test (see Figure 6.1). In the pre-test participants were asked to classify their level of agreement with a series of phrases (likert scale ranging from -2 to 2). Each phrase corresponded to a different feature: awareness, behavior understandability, personality, visual impact, predictability, friendliness, intelligence, likability and interest (see Table 6.1). These phrases were iteratively constructed during the Preliminary Tests, with the final versions achieving some consensus concerning their understandability.

In the post-test participants were again asked to classify their level of agreement with the same phrases. Values in the post-test were compared to values in the pre-test, with a new variable being created for each feature, except for predictability. These variables had three possible values: -1 if the participant agreement score decreased from pre-test to post-test; 0 if the agreement scores in pre-test and post-test were the same; and 1 if the post-test agreement score was higher than the pre-test one. Subsequent analyses of awareness, behavior understandability, personality and visual impact was done using these variables. Predictability will be treated separately due to its exceptional nature.

Table 6.1: Pre-test and Post-test phrases

feature	phrase
believability features	
awareness	Meemos perceive the world around them.
behavior understandability	It is easy to understand what a Meemo is thinking about.
personality	Meemos have personalities.
visual impact	Meemos' behavior draws my attention.
predictability	Meemos' behavior is predictable.
video game features	
friendliness	Meemos are friendly creatures.
intelligence	Meemos are intelligent creatures.
likability	I like Meemos.
interest1	Meemos' behavior is interesting

In addition, there were three phrases in the "Act 2 Questions" not present in the pre-test, but that were also iteratively created during the Preliminary Tests. These phrases concerned: behavior coherence, change with experience and interest (see Table 6.2). As interest is considered in two questions, the values obtained for the first phrase will be referred to as interest1 and the values for the additional phrase as interest2.

Table 6.2: Additional Post-test Phrases

feature	phrase
behavior coherence	Meemos' behavior is coherent.
change with experience	Meemos' behavior changes according to experience.
interest2	I would like to see more of Meemos in the future.

6.2.6 Results

To help us understand which type of test should be chosen to analyze the differences between conditions, the normality of the feature variables was analyzed. We used the Kolmogorov-Smirnov test, with the Lilliefors Significance Correction since the average and variance of the population were unknown. Each variable, factorized by test condition, presented a significant Kolmogorov-Smirnov value ($p < 0.001$) which represented a clear non-normality in all cases. This non-normality, together with the fact that variables were ordinal, led us to choose a non-parametric test. Consequently, we proceeded with a series of Kruskal-Wallis tests to compare the three conditions (2 degrees of freedom). The values of these tests are presented in Table 6.3.

From the tests' results, we can see that all features were significantly affected by the test condition ($p < 0.05$)[3], apart from visual impact. Therefore we exclude visual impact from further analyses. This exception was probably due to the fact that the videos were very focused on the agents' actions. With nothing else to draw the attention of participants, the values for visual impact were similar for all test conditions.

Additionally, by reviewing the descriptive statistics also present in Table 6.3, one can already identify some differences between test conditions. Comparing *retrieval* and *no retrieval*, *retrieval*

[3]Tests presented use Fisher's criterion for alpha-error: 0.05[18](pp. 149–153).

Table 6.3: Kruskal-Wallis for features (2 degrees of freedom)

features	Descriptive Statistics			Kruskal-Wallis differences between conditions
	no retrieval (N=31) Mdn[Quartiles]	retrieval (N=30) Mdn[Quartiles]	random expression (N=29) Mdn[Quartiles]	
behavior coherence	0[-1,1]	1[1,2]	1[0,1]	$\chi^2 = 23.655$ $p < 0.001$
change with experience	0[-1.5,0]	1[1,2]	1[1,2]	$\chi^2 = 40.371$ $p < 0.001$
awareness	0[-1,0]	0[0,1]	0[0,1]	$\chi^2 = 18.053$ $p < 0.001$
behavior understandability	0[-1,0]	0[0,1]	0[-1,0]	$\chi^2 = 13.838$ $p = 0.001$
personality	-1[-1,0]	0[0,0]	0[0,1]	$\chi^2 = 17.436$ $p < 0.001$
visual impact	0[-1,0]	0[0,0]	0[0,1]	$\chi^2 = 4.241$ $p = 0.120$
friendliness	0[-1,0]	0[0,0]	0[-1,0]	$\chi^2 = 7.549$ $p = 0.023$
intelligence	0[-1,0.5]	1[0,1]	0[0,1]	$\chi^2 = 9.607$ $p = 0.008$
likability	0[-1,0]	0[0,0]	0[-1,0]	$\chi^2 = 9.901$ $p = 0.007$
interest1	0[-1,0]	0[0,1]	0[0,1]	$\chi^2 = 12.196$ $p = 0.002$
interest2	0[1.5,1]	1[0,1]	0[0,1]	$\chi^2 = 8.281$ $p = 0.016$

appears to present higher values for all the features. On the other hand, if we compare *retrieval* and *random expression* one can recognize that for some features the values are similar, for others *retrieval* presents higher values, and in the specific case of the personality feature *random expression* appears to have higher values. This was probably caused by two reasons: participants finding the agent's behavior in this test condition unexpected; and some participants finding meemo 1 "mean", or even "sadistic" (words taken form the justification of the agent's behavior), which led them to interpret its behavior as unique. This was a consequence of the expression chosen for test condition *random expression* (*happiness*), but we will discuss this further on.

We used several Mann-Whitney tests to follow up these findings. Two sets of comparisons were made: comparing test condition *retrieval* with test condition *no retrieval*; and comparing test condition *retrieval* with test condition *random expression*. As there were two comparisons we used the Bonferroni correction, so instead of Fisher's 0.05 criterion, we considered half of this value (0.025). On the other hand we assumed that the values for test condition *retrieval* would be higher than values for test condition *no retrieval*. Similarly, we assumed that the values for test condition *retrieval* would be higher than values for test condition *random expression*, except in the personality feature for which we considered they would be lower. Given our assumptions, we always used one-tailed tests.

Retrieval versus *No Retrieval*

The results for comparing the features between test condition *retrieval* and *no retrieval* are presented in Table 6.4. As can be seen in the results, participants had a significantly ($p < 0.025$) enhanced perception of the believability features in test condition *retrieval* compared to test condition *no retrieval*, with the effect of these features accounting for 22% to 50% of the variance. Considering these results, *no retrieval* appears to have been perceived as less believable than test condition *retrieval*, which is consist with our hypothesis. In addition, agents in test condition *retrieval* were also considered more friendly, more intelligent, more likable and more interesting, with significant differences ($p < 0.025$) and the effect of these features accounting for 9% to 15% of the variance.

Table 6.4: Mann-Whitney - *retrieval* versus *no retrieval* and *retrieval* versus *random expression*

features	differences between *retrieval* and *no retrieval* (one-tailed)	differences between *retrieval* and *random expression* (one-tailed)
behavior coherence	$U = 160.0$, $p < 0.001$, $r = -0,625$	$U = 291.5$, $p = 0.003$, $r = -0.347$
change with experience	$U = 110.0$, $p < 0.001$, $r = -0.705$	$U = 490.5$, $p = 0.185$, $r = -0.115$
awareness	$U = 216.5$, $p < 0.001$ $r = -0.481$,	$U = 433.5$, $p = 0.297$, $r = -0.068$
behavior understandability	$U = 272.0$, $p < 0.001$, $r = -0.466$	$U = 359.0$, $p = 0.041$, $r = -0.222$
personality	$U = 253.0$, $p < 0.001$, $r = -0.474$	$U = 458.0$, $p = 0.452$, $r = -0.015$
friendliness	$U = 373.0$, $p = 0.007$, $r = -0.305$	$U = 325.0$, $p = 0.007$, $r = -0.313$
intelligence	$U = 308.0$, $p = 0.001$, $r = -0.387$	$U = 344.5$, $p = 0.028$, $r = -0.245$
likability	$U = 330.5$, $p = 0.001$, $r = -0,391$	$U = 316.5$, $p = 0.009$, $r = -0,307$
interest1	$U = 348.0$, $p = 0.004$, $r = -0.327$	$U = 421.5$, $p = 0.234$, $r = -0.093$
interest2	$U = 324.0$, $p = 0.003$, $r = -0.344$	$U = 376.0$, $p = 0.090$, $r = -0.172$

Retrieval versus *Random Expression*

If we turn to the comparison between test condition *retrieval* and *random expression* (results also in Table 6.4), we see that only one believability feature was significantly different ($p < 0.025$): behavior coherence (its result's box plot is presented in Figure 6.2). The similar results for change with experience, awareness and behavior understandability have several causes.

Concerning test condition *random expression*, many participants suggested that the *happiness* reaction of meemo 1 was due to the fact that it avoided the trap ("Since Meemo 1 did not fall in the trap, he reacted with happiness"). This interpretation induced a false sense of awareness of the trap. Other interpretations involved meemo 1 having learned where the trap was, or being relieved about not being caught in the trap, with explicit or implicit references to memory. These interpretations contributed to the sense of meemo 1 changing with experience for test condition *random expression*. In general it was easy for participants to create a justification for the emotional reaction of meemo 1, which led to behavior understandability values

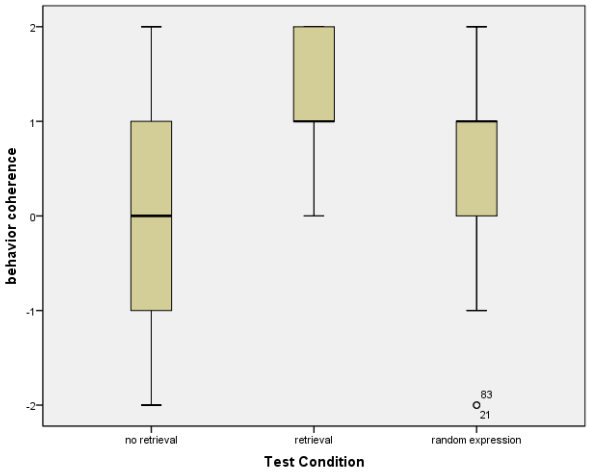


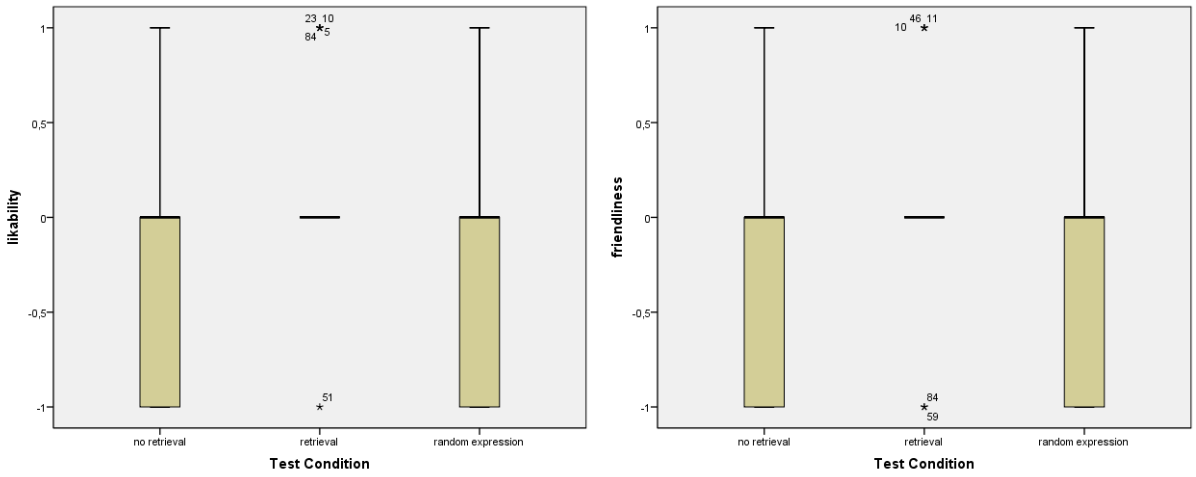
Figure 6.2: Box plot for behavior coherence

not being significantly different from the ones of test condition *retrieval*. One possible reason for the ease in justification might have been the lack of detail in the scenario's description: with few information, participants were free to create their own interpretation of the story.

In addition, values for behavior understandability, as well as for change with experience and awareness, were greatly influenced by how we chose to simulate the system represented by test condition *random expression*. We were simulating a system in which agents would express emotions in a randomized sequence. These would result in four different situations in which the agent expressed these randomized emotions. We chose situation III because it fitted best in our test scenario. In situation III the agent has an emotional reaction when an agent driven by our architecture would, but the emotion expressed is different. The problem is that situation II (the agent expressing an emotion in a location where nothing has happened to him) can potentially happen much more often than situation III. Therefore, we believe that if a participant was exposed to a longer simulation of the test condition *random expression*, the values for change with experience, awareness and behavior understandability would decrease.

On the other hand values for behavior coherence were significantly higher for test condition *retrieval* ($p < 0.025$) with an effect accounting for 12% of the variance. Ortony [42] pointed out the importance of behavior coherence for believability, and how it helps an individual to create a mental model of the character's behavior. Consequently, this result is in line with our hypothesis. Nonetheless, participants would only get a clearer sense of agent's coherence after a longer exposure to their behavior. Still concerning believability features, it should be noted that although the personality feature values were slightly higher for test condition *random expression* as mentioned before (see Table 6.3), this difference was not significant.

Lastly, in what concerns video game features, only likability and friendliness presented significantly ($p < 0.025$) different values (box plots are shown in Figure 6.3). The effect of likability accounts for 9% of the variance and the friendliness one accounts for 10%. These results are coherent with participant's justifications of the behavior of meemo 1 in test condition *random expression*: many found the agent "mean", "sadistic", and expressed dislike or even disgust towards it. It is curious to see that these perceptions of the agent had nothing to do with the main design decision determining the agent's character: the reaction rule. Implicitly, by determining



(a) likability

(b) friendliness

Figure 6.3: Box plots for likability and friendliness

that agents perceive as undesirable seeing an agent fall in a hole, we were encoding that agents were friendly towards each other. Furthermore, the agents' graphical model was designed to be cute, with a large head in proportion with the whole body [47] and huge eyes for a baby like face [28] (pp. 10–12). These attributes can be seen as other design decisions that were contradicted by the participant's perception of agents in test condition *random expression*.

However, we again point out that these contradicting perceptions depended of the emotion chosen for test condition *random expression*. Had we chosen an angry expression, we suspect that meemo 1 would not cause the mentioned dislike feelings. Nevertheless, the test condition exemplified a possible situation: the emotion generated having a different valence from the one generated by our architecture.

6.2.7 Predictability Results

Purposefully, we have yet mentioned the predictability results for the different test conditions. While for other believability features higher values correspond to an enhanced sense of believability, in predictability very high values, as well as very low ones, correspond to a decreased sense of believability. For each participant, we compared the value in the pre-test to the value in the post-test, by subtracting the first to the second. All subsequent analyses were done using this calculated differences. We can see some descriptive statistics concerning these differences in Figure 6.4.

One notices that the results for test condition *retireval* are higly centered in 0 (71% actually being 0). Thus exposure to test condition *retrieval* does not seem to have affected considerably participant's sense of agents predictability. Consequently, in this scenario the model did not harm believability through predictability. However, similarly to behavior coherence, more solid results can only be obtained if the scenario was longer and had more emotionally relevant episodes.

One also notices that agents in the other two conditions appear to have been perceived as less predictable. However the comparison between conditions, in what concerns predictability's

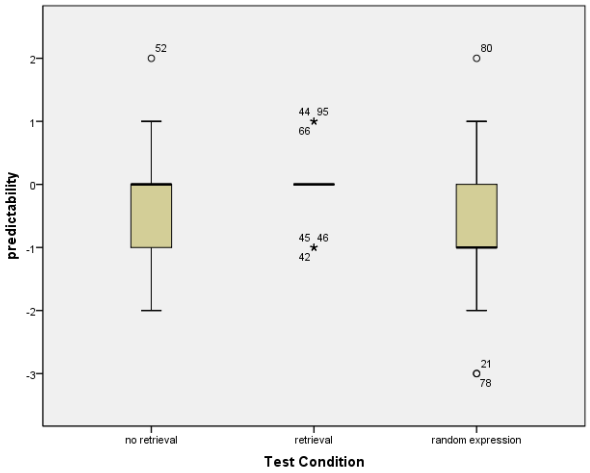


Figure 6.4: Box plot for predictability

influence on believability, could only be done with a finer grained scale. For instance, the scale could have nine possible values, five of which would be: very predictable, predictable, nor predictable or unpredictable, unpredictable, and very unpredictable.

6.3 Concluding Remarks

In this chapter we started by describing the some preliminary tests of our agent architecture integrated with the developed application. Participants were presented with variations of the complete scenario described in Implementation. They found the scenario too complex, had difficulty in seeing agent's facial expressions and considered the experiment's duration adequate.

Subsequently, we described the final evaluation in which these points were accounted. We started by justifying why we opted by a non-interactive experiment due to time-line and resource constraints. Then we classified our group of 96 participants as being adults with a relatively balanced gender distribution. Further on we described how participants were exposed to a simple story in which character's behavior was driven by our architecture. To test our hypothesis we manipulated our test with three test conditions: *retrieval*, representing our architecture, and two other representing variations of our architecture without episodic memory retrieval (*no retrieval* and *random expression*). In an effort to do an objective analysis of believability we indirectly measured it through believability features. Additionally, we also analyzed features that can be used to classify game characters (video game features).

First of all, visual impact was not significantly different between test conditions, probably because in the experience participants' attention was completely focused on meemos. Secondly, when analyzing the values of behavior coherence, change with experience, awareness, behavior understandability and personality, we realized they were significantly higher for test condition *retrieval* than for test condition *no retrieval*. Moreover, they accounted for 22% to 50% of the variance. On the whole results seem to indicate that test condition *retrieval* was perceived as more believable than test condition *no retrieval*. This conclusion is consistent with our hypothesis.

Turning to the comparative analysis with test condition *random expression*, for change with

experience, awareness and behavior understandability test condition *retrieval* did not present significantly higher values. We believe that the two main contributing factors for this were: the scenario's description not being very detailed thus allowing a wide range of interpretations; the type (III) of the situation simulated for test condition *random expression*.

On the other hand, test condition *retrieval* presented significantly higher values of behavior coherence than test condition *random expression*. Behavior coherence accounted for 12% of the variance, and being an important factor for an enhanced sense of believability (as described by Ortony [42]), these results are also consistent with our hypothesis. We also realized that exposure to test condition *retrieval* did not seem to considerably affect participants perception of predictability. Thus, not affecting believability through predictability.

Additionally, when analyzing the video game features, we identified that the likability values were significantly lower for test condition *random expression*. Furthermore, some participants found meemo 1, in this test condition, to be "evil", "mean" and even "sadistic". All these perceptions conflict with the meemo's design: reaction rule implying agreeableness; and graphical model designed to be cute. However, this was a consequence of the emotion expression chosen for *random expression*. Last, we did some secondary analyses that can be consulted in Appendix B.

Summing up, test condition *retrieval* was perceived as more believable than test condition *no retrieval*, which is consistent with our hypothesis. When comparing test condition *retrieval* and *random expression*, *retrieval* presented significantly higher values for behavior coherence, which is also consistent with our hypothesis. Nonetheless to analyze this hypothesis properly, further testing needs to be performed. In particular with a longer scenario in which agents are faced with a wider range of emotionally relevant episodes.

Chapter 7

Conclusion

We have argued that episodic memory retrieval is an important element of human experience, and as such should be considered when developing intelligent believable agents. Literature from the area of the cognitive sciences and several concepts of believability were considered when defining the architecture's background. We analyzed several architectures for autonomous agents that had some of our architecture's requirements. None, however, fulfilled all of them: supporting episodic memories; memories having associated emotional content; focus on agent believability; supporting emotional appraisal; modelling ecphory; modelling recollective experience.

We presented a possible solution to the thesis problem by describing a model for agent episodic memory retrieval divided in two steps: ecphory and recollective experience. Ecphory consists of calculating a perceptual distance, for each episodic memory trace, between the perceived stimuli (retrieval cues) and the perceptual memory (perceived stimuli when the memory trace was stored). Memory traces for which this distance is small are selected for the recollective experience. We also described the location ecphory approximation, in which locations are interpreted as indirect retrieval cues. Then we presented how a generic appraisal model could be adapted to serve as a recollective experience model.

Putting these concepts into practice, we implemented an agent architecture supporting them. Our location ecphory implementation used the euclidean distance, and our recollective experience was a reactive appraisal module that had reaction rules for retrieval events. This architecture was used to model character's behavior in a game application. Among other externalizations of the though process, characters express their most intense emotion through facial expressions. It was this application that supported our evaluation, albeit being a non-interactive one.

The 96 participants of the final evaluation were randomly distributed by three test groups, two of which were control ones. One control group was exposed to a simulated variation of our architecture without episodic memory retrieval (*no retrieval*). The other control group was exposed to a simulated variation without episodic memory retrieval, but with random expression of emotions (*random expression*). Believability differences between conditions were evaluated indirectly through believability features. To assess them, we performed a series of between-groups tests with a pre-test/post-test structure.

Agents of *no retrieval* were considered less believable than agents modeled in our architecture, which is consistent with our hypothesis. Compared to *random expression* agents, the behavior coherence perception was more intense for our architecture's agents, which is also consistent

with the hypothesis.

Nonetheless, to analyze it properly, further testing needs to be performed. In particular with a longer scenario in which agents are faced with a wider range of emotionally relevant episodes. This would require the creation of more content for the application: objects, reaction rules, events, levels, etc. Moreover, the architecture should support more emotion types, and all emotions should have an externalization. This last point would probably require the emotion expression to be extended to body positioning and movement.

Additionally, it would also be interesting to evaluate the model in a scenario in which participants were allowed to directly interact with agents. To do this, using the already developed application, some aspects would need to be addressed: camera positioning should be handled automatically maximizing agent's face visibility; walls should be made transparent according to camera positioning, also to maximize face visibility; the interface for commanding minions should be simplified; some game menus would need to be created.

Concerning the model itself, and trying to avoid the agent's behavior becoming predictable during a longer interaction, an element of variability could be added to it. For instance, one could associate a stochastic process to the ecphory model, with the probability of a memory trace being elicited depending on the perceptual distance. Another possibility, would be in the location ecphory approximation, instead of considering a simple radius, using an approximately circular area but with an irregular border (introducing noise). Furthermore, some mechanism of memory storage forgetfulness would need to be added to the architecture, or else in longer interactions the memory needed might exceed hardware requirements.

Finally, it would be interesting to model individual retrieval cues in ecphory, as well as considering different distance functions. Further differentiation of re-appraisal of past events, compared to the original experience of events, should be modeled, maintaining the inspiration in cognitive sciences research. To conclude, we believe our work represents a small step, yet relevant, towards modelling memory retrieval on agents and analyzing its impact on agent believability.

Bibliography

[1] F. Attneave. Dimensions of similarity. *The American Journal of Psychology*, 63(4):516–556, 1950.

[2] J. Bach. The micropsi agent architecture. In *Proceedings of ICCM-5*, Universitäts–Verlag Bamberg, 2003.

[3] C. R. Barclay. *Schematization of autobiographical memory*, chapter 6, pages 82–99. Press, Cambridge University, 1988.

[4] J. Bates. The role of emotion in believable agents. *Commun. ACM*, 37(7):122–125, 1994. 176803.

[5] L. M. Botelho and H. Coelho. A schema-associative model of memory. In *Golden West International Conference on Intelligent Systems (GWICS'95)*, pages 81–85, Raleigh, NC, USA, 1995. International Society for Computers and their Applications (ISCA).

[6] L. M. Botelho and H. Coelho. Machinery for artificial emotions. *Cybernetics and Systems*, 32(5):465–506, 2001.

[7] C. Brom and J. Lukavsky. *Intelligent Virtual Agents*, chapter Towards More Human-Like Episodic Memory for More Human-Like Agents, pages 484–485. Springer, 2009.

[8] C. Brom, K. Pešková, and J. Lukavsky. What does your actor remember? towards characters with a full episodic memory. In *ICVS'07: Proceedings of the 4th international conference on Virtual storytelling*, pages 89–101, Berlin, Heidelberg, 2007. Springer–Verlag.

[9] L. D. Canamero and J. Fredslund. How does it feel? emotional interaction with a humanoid lego robot. *Socially Intelligent Agents: The Human in the Loop. Papers from the AAAI 2000 Fall Symposium*, pages 23–28, 2000.

[10] S. T. Coleridge. *Biographia Literaria: Biographical Sketches of my Literary Life & Opinions*. Princeton University Press, 1985.

[11] S. T. Coleridge. *Lyrical Ballads*. Penguin Classics, 2007.

[12] A. M. Colman. *Dictionary of Psychology*. Oxford University Press, third edition, 2009.

[13] J. Dias. Fearnot!: Autonomous synthetic characters for emphatic interactions. Master's thesis, Universidade Técnica de Lisboa - Instituto Superior Técnico, 2005.

[14] J. Dias, W. Ho, T. Vogt, N. Beeckman, A. Paiva, and E. André. I know what i did last summer: Autobiographic memory in synthetic characters. In *Affective Computing and Intelligent Interaction*, pages 606–617, 2007.

[15] M. S. El-Nasr, J. Yen, and T. R. Ioerger. Flame—fuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-Agent Systems*, 3(3):219–257, 2000.

[16] C. D. Elliott. *The affective reasoner: a process model of emotions in a multi-agent system.* PhD thesis, Northwestern University, 1992.

[17] P. C. Ellsworth and K. R. Scherer. *Appraisal Processes in Emotion*, chapter 29. Oxford University Press US, 2003.

[18] A. Field and G. Hole. *How to Design and Report Experiments*, chapter Experimental Designs, pages 54–84. Sage, 2003.

[19] N. H. Frijda. *The emotions.* Cambridge University Press, 1986.

[20] J. Gratch. Émile: Marshalling passions in training and education, 2000.

[21] J. Gratch and S. Marsella. Tears and fears: modeling emotions and emotional behaviors in synthetic agents, 2001.

[22] *Collins English Dictionary – Complete and Unabridged.* HarperCollins Publishers, 2003.

[23] W. C. Ho. *Computational memory architectures for autobiographic and narrative virtual agents.* PhD thesis, School of Computer Science - University of Hertfordshire, 2005.

[24] W. C. Ho, K. Dautenhahn, and C. L. Nehaniv. Computational memory architectures for autobiographic agents interacting in a complex virtual environment: a working model. *Connection Science*, 20(1):21–65, 2008.

[25] W. C. Ho, J. Dias, R. Figueiredo, and A. Paiva. Agents that remember can tell stories: integrating autobiographic memory into emotional agents, 2007. 13291381-3.

[26] W. C. Ho and S. Watson. *Intelligent Virtual Agents*, chapter Autobiographic Knowledge for Believable Virtual Characters, pages 383–394. Springer Berlin / Heidelberg, 2006.

[27] *The American Heritage Dictionary of the English Language.* Houghton Mifflin Company, fourth edition, 2009.

[28] K. Isbister. *Better Game Characters by Design - A Psychological Approach.* Morgan Kaufmann Publishers, 2006.

[29] C. Izawa. *On Human Memory: Evolution, Progress, and Reflections on the 30th Anniversary of the Atkinson-Shiffrin Model.* Lawrence Erlbaum Associates, 1999.

[30] N. Kapur. Syndromes of retrograde amnesia: A conceptual and empirical synthesis. *Psychological Bulletin*, 125(6):800–825, 1999.

[31] J. Lasseter. Principles of traditional animation applied to 3d computer animation. *SIGGRAPH Comput. Graph.*, 21(4):35–44, 1987.

[32] J. C. Lester and B. A. Stone. Increasing believability in animated pedagogical agents, 1997.

[33] A. B. Loyall. *Believable Agents: Building Interactive Personalities.* PhD thesis, Carnegie Mellon University, 1997.

[34] S. Marsella and W. Johnson. *Intelligent Tutoring Systems*, volume 1452/1998 of *Lecture Notes in Computer Science*, chapter An Instructor's Assistant for Team-Training in Dynamic Multi-agent Virtual Worlds, pages 464–473. Springer Berlin / Heidelberg, 1998.

[35] S. C. Marsella, W. L. Johnson, and C. LaBore. Interactive pedagogical drama, 2000.

[36] C. Martinho. Emotions in motion: short time development of believable pathematic agents in intelligent virtual environments. Master's thesis, Universidade Técnica de Lisboa - Instituto Superior Técnico, 1999.

[37] C. Martinho. Cglib, 2007.

[38] A. R. Mayes and N. Roberts. Theories of episodic memory. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 356(1413):1395–1408, 2001.

[39] G. A. Miller. *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information*, chapter 23, pages 357–372. MIT Press, 2003.

[40] L. Morgado and G. Gaspar. Towards background emotion modeling for embodied virtual agents. In *AAMAS '08: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems*, pages 175–182, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

[41] K. Oatley, D. Keltner, and J. M. Jenkins. *Understanding Emotions*, chapter Communication of Emotions, pages 83–114. Blackwell Publishing, 2006.

[42] A. Ortony. *Emotions in Humans and Artifacts*, chapter On making believable emotional agents believable. MIT Press, 2003.

[43] A. Ortony, G. L. Clore, and A. Collins. *The Cognitive Structure of Emotions*. Published by Cambridge University Press, 1990.

[44] J. Pantaleão, A. Catarrinho, P. Branco, and P. Gomes. Meemo rescue, 2008.

[45] E. Phelps and T. Sharot. How (and why) emotion enhances the subjective sense of recollection. *Current Directions in Psychological Science*, 2008.

[46] R. W. Picard. *Affective computing*, chapter Affective Signals and Systems, pages 141–164. MIT Press, 1997.

[47] A. Rollings and E. Adams. *Andrew Rollings and Ernest Adams on Game Design*, chapter Character Development. New Riders Games, 2003. 1213088.

[48] I. J. Roseman and C. A. Smith. *Appraisal Theory*, chapter 1, pages 3–16. Oxford University Press, 2001.

[49] M. D. Rugg and E. L. Wilding. Retrieval processing and episodic memory. *Trends in Cognitive Sciences*, 4(3):108–115, 2000.

[50] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, chapter Informed Search and Exploration, pages 94–136. Prentice-Hall, Inc., 1995.

[51] S. Santini and R. Jain. Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:871–883, 1999.

[52] F. Thomas and O. Johnston. *Disney Animation: The Illusion of Life*. Abbeville Press, New York, 1981.

[53] E. Tulving. Episodic memory: From mind to brain. *Annual Review of Psychology*, 53:1–25, 2002.

[54] E. Tulving, M. E. L. Voi, D. A. Routh, and E. Loftus. Ecphoric processes in episodic memory [and discussion]. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences (1934-1990)*, 302(1110):361–371, 1983.

[55] M. A. Wheeler and C. T. McMillan. Focal retrograde amnesia and the episodic-semantic distinction. *Cognitive, Affective, & Behavioral Neuroscience*, 1(1):22–37, 2001.

[56] J. T. Wixted. The psychology and neuroscience of forgetting. *Annual Review of Psychology*, 55(1):235–269, 2004.

Appendix A

Experiment's Questionnaire

Meemos

Thank you for your help! This anonymous experiment takes about 10 minutes. You will see a few short videos and answer some questions about them. Please read carefully the following text.

Introduction

Imagine a fantasy world populated by creatures called **meemos**(Figure1).

Figure1 - A meemo

Hunter is someone that eats meemos. He puts traps in the world to catch meemos. Video0 shows a meemo being caught by a **Trap**. Click on the link to see it. The video will loop automatically. Video0 Link

Traps are really deep holes. When a meemo reaches the bottom, he dies. The Hunter never puts traps close to each other.

Two meemos, that we will call **Meemo1** and **Meemo2**, are currently going through a **Tunnel**(Figure2) that has a Trap. They don't know about the Trap. Meemo2 is going to be caught in the Trap.

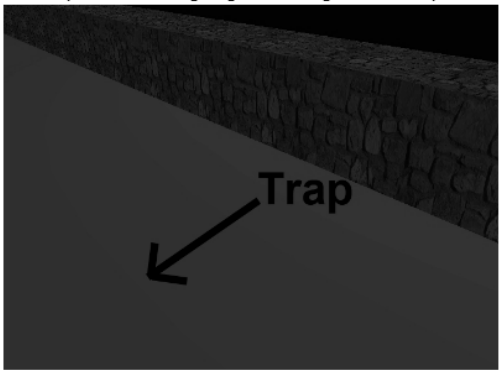


Figure2 - Tunnel with Trap

Expectations

You will now be asked some questions regarding your current expectations towards meemos.

Please classify the following sentences according to your level of agreement with them.
(-2: I totally disagree with the sentence; 0: I don't agree or disagree with the sentence; 2: I totally agree with the sentence)

Meemos are friendly creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos are intelligent creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos have personalities.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
It will be easy to understand what a Meemo is thinking about.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior will draw my attention.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos perceive the world around them.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
I will like Meemos.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2

Video1

Please click on the link bellow to see Video1. Watch it carefully.
The video will automatically start to loop after it ends. Video1 Link

Please classify the emotional state of the two meemos before and after Meemo2 was caught in the Trap.
(select one or more emotional state/states for each question)

How did Meemo1 feel before Meemo2 was caught in the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy
How did Meemo1 feel after Meemo2 was caught in the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy
How did Meemo2 feel before he was caught in the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy
How did Meemo2 feel after he was caught in the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy

Please classify the following sentences according to your level of agreement with them.
(-2: I totally disagree with the sentence; 0: I don't agree or disagree with the sentence; 2: I totally agree with the sentence)

Meemos are friendly creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos are intelligent creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos have personalities.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
It is easy to understand what a Meemo is thinking about.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior draws my attention.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos perceive the world around them.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
I like Meemos.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior is interesting.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior is predictable.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2

After you clicked next, please don't go back to this page to change your answers.

Next >>

Meemos

Video2

Please read the following text.

Meemo2 was caught in the Trap and died. After some time, Meemo1 uses the Tunnel where Meemo2 was caught.

Click on the link bellow to see Video2. Watch it carefully.
The video will automatically start to loop after it ends. Video2 Link

Please classify the following sentences according to your level of agreement with them.
(-2: I totally disagree with the sentence; 0: I don't agree or disagree with the sentence; 2: I totally agree with the sentence)

Meemos are friendly creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos are intelligent creatures.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos have personalities.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
It is easy to understand what a Meemo is thinking about.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior draws my attention.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos perceive the world around them.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
I like Meemos.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior is interesting.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior is predictable.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2

Please classify the emotional state of Meemo1 before and after he was close to the Trap.
(select one or more emotional state/states for each question)

How did Meemo1 feel before being close to the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy
How did Meemo1 feel after being close to the Trap?	☐ Angry ☐ Disgusted ☐ Fearful ☐ Sad ☐ Neutral ☐ Surprised ☐ Happy

Please classify the following sentence according to your level of agreement with it.
(-2: I totally disagree with the sentence; 0: I don't agree or disagree with the sentence; 2: I totally agree with the sentence)

I expected an emotional reaction from Meemo1.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2

Please classify the following sentences according to your level of agreement with them.
(-2: I totally disagree with the sentence; 0: I don't agree or disagree with the sentence; 2: I totally agree with the sentence)

I would like to see more of Meemos in the future.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior changes according to experience.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2
Meemos' behavior is coherent.	○ -2 ○ -1 ○ 0 ○ 1 ○ 2

Why do you think Meemo1 behaved as he did in Video2?

Conclusion

How familiar are you with video games?
(-2: not familiar at all; 0: more or less familiar; 2: very familiar)

○ -2 ○ -1 ○ 0 ○ 1 ○ 2

How familiar are you with notion of believability in the context of Computer Science?
(-2: not familiar at all; 0: more or less familiar; 2: very familiar)

○ -2 ○ -1 ○ 0 ○ 1 ○ 2

Gender:	○ Male ○ Female
Age:	○ Less than 14 ○ Between 14 and 18 ○ Between 19 and 23 ○ Between 24 and 28 ○ Between 29 and 38 ○ Between 39 and 48 ○ More than 49
Country:	
How would you classify your english reading skills?	○ inexistent ○ poor ○ fair ○ good ○ excellent

Your comments and additional feedback are more than welcome. You can leave them in the text box bellow.

If you want to receive feedback about the experiment results, put your email address in the text box bellow.

<< Back Send

This form was created at www.formdesk.com

Appendix B

Secondary Evaluation Analyses

Besides the main tests presented in the Evaluation chapter, we have also done some additional analyses. We considered how well did participants identify agents' emotional expressions, we analyzed wich emotional reactions did participants expect in "Act 2", and assessed the possible influence of emotional expressions alone in believability. We proceed by describing these three analyses.

B.1 Emotion Identifiability

We had identified in the preliminary tests that participants were having difficulty identifying agents' emotional expressions. To validate that this problem had been overcome in the final evaluation, we considered which emotions did they identify. Participants were asked to identify which emotional expressions they had witnessed in "Act 1" and in "Act 2". They were given the choice between the following emotional states: anger, disgust, fear, sadness, neutral, surprise and happiness. The recognition rates are presented in Table B.1. Participants were allowed to select more than one emotion, and only if the participant selected solely the emotion displayed was it considered a valid recognition.

Table B.1: Emotion identifiability

agent	situation	emotion displayed	recognition rate
meemo 1	Act 1 - before trap was activated	neutral	92%
meemo 2	Act 1 - before trap was activated	neutral	92%
meemo 1	Act 1 - after trap was activated	sadness	79%
meemo 2	Act 1 - after trap was activated	sadness	74%
meemo 1	Act 2 - before being close to the trap	neutral	87%
meemo 1	Act 2 - after being close to the trap	neutral	91%
		sadness	74%
		happiness	97%

We compared the recognition rates of our application's emotional expressions, with the recognition rates achieved in a experiment performed by Canamero [9]. In Canamero's experiment the emotional facial expressions of a robot (Felix) were evaluated. These emotional expressions depended only on mouth and eyebrow position and shape, in similarity with our application. Participants were presented with a sequence of five of Felix's expressions: anger, sadness, fear, happiness and surprise. For each emotion they were asked to choose from a set of nine options: anger, sadness, fear, happiness, surprise, disgust, anxiety, pride and worry. We considered the recognition rates calculated for adults (ages ranging between 15–57). The recognition rates in our experiment ranged from 74% to 97% and in Cabamero's experiment, for the same emotions,

the recognition rates ranged from 64% to 81%. As our recognition rates do not seem to be considerably lower, we believe that the emotion identifiability was adequate.

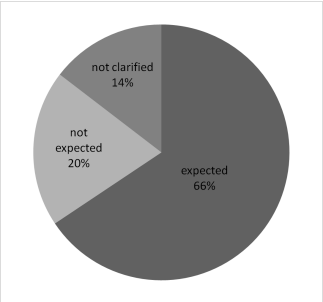
B.2 Participant's Expectations

Believability has to do with users, or observers, identifying an agent's goals, beliefs and personality [32]. In this process expectations are created. Therefore we were interested in analyzing what emotional reaction did participants expect in "Act 2".

First of all, participants were asked (in "Act 2 Questions") if they expected meemo 1 to emotionally react when walking close by the trap the second time. Analyzing the answer frequencies we identified that 66% of the participants claimed expecting a reaction, 20% claimed not expecting a reaction and 14% did not clarify their opinion (Table B.2).

Subsequently, participants that indicated expecting a reaction were asked which emotion did they expect to be expressed. A percentage of 37% selected two or more emotions. In virtue of this, we considered all participants that selected an emotion, even if some selected others as well. The percentages for each of the expected emotions are presented in Table B.2. As can be seen, the most selected emotional reaction was sadness (60%), this being the one selected by our model. Nonetheless, there was also a considerable amount of participants selecting fear (40%). This might indicate that an architecture supporting episodic memory retrieval trough the reappraisal of past events should also support expectation related emotions.

Table B.2: Emotion expectations in "Act 2 Questions"



emotion	percentage
sadness	60%
fear	40%
happiness	16%
angry	16%
disgust	9%
surprise	7%

B.3 Emotional Expressions' influence on Believability and Video Game Features

It could be argued that some of the differences between test conditions in the Evaluation's results depended solely on the emotional expressions displayed. This would be the case if an agent displaying a certain expression, independent of the situation, influenced significantly the perception of one of the analyzed features.

In order to deal with this problem the experiment had a second manipulation and the question section "Expressions Questions". In the "Introduction" participants saw a picture of an agent. The picture showed a meemo in the same position with one of three expressions: *sadness* (test condition *sadness expression*), *neutral* (test condition *neutral expression*) and *happiness* (test condition *happiness expression*). This part of the experiment was embedded in the introduction so that participants would not realize that they were evaluating features simply based on facial expressions.

There were no questions concerning behavior coherence, change with experience, predictability and interest. We decided not to test these features in this phase because they inherently require more exposure to the agents in order to be correctly evaluated. In the preliminary tests we noticed that, specially for the feature behavior coherence, an early question would be answered by wild guessing and it perturbed participants to do so.

In addition, to minimize the increase of variance in the remaining results, test conditions in the "Introduction" manipulation were mapped to test conditions of the main tests. A participant that was exposed to test condition *sadness expression* was also exposed to test condition *retrieval*. A participant that was exposed to test condition *neutral expression* was subsequently exposed to test condition *no retrieval*. Lastly, test condition *happiness expression* was mapped to test condition *random expression*.

We started by analyzing the normality of the feature values for section "Expressions Questions". We used a series of Kolmogorov-Smirnov tests with the Lilliefors Significance Correction since the averages and variances of the population were not known. Each feature factorized by the test condition presented a significant Kolmogorov-Smirnov value ($p < 0.05$). Consequently the variables were clearly not normally distributed. This non-normality, together with the fact that the variables were ordinal, led us to chose a non-parametric test. Thus we proceeded with several Kruskal-Wallis tests to evaluate if the features were affected by the manipulation (results in Table B.3). With the exception of friendliness, all other features were not significantly different ($p > 0.05$) between test groups.

Table B.3: Kruskal-Wallis for features in "Expressions Questions" (2 degrees of freedom)

features	Descriptive Statistics			Kruskal-Wallis differences between conditions
	neutral expression (N=31) Mdn[Quartiles]	*sadness expression* (N=30) Mdn[Quartiles]	*happiness expression* (N=29) Mdn[Quartiles]	
awareness	0[-2,1]	0[0,1]	0[-1,1]	$\chi^2 = 1.505$ $p = 0.471$
behavior understandability	0[-1,1]	0[-2,0]	0[-1,0]	$\chi^2 = 2.339$ $p = 0.311$
personality	0[-1,1]	0[0,1]	0[-1,1]	$\chi^2 = 1.894$ $p = 0.388$
visual impact	0.5[-1,1]	1[0,1]	1[0,1]	$\chi^2 = 1.179$ $p = 0.555$
friendliness	1[0,1]	0[0,1]	1[1,2]	$\chi^2 = 9.324$ $p = 0.009$
intelligence	-1[-1,0]	-1[-1,0]	-1[-1,0]	$\chi^2 = 0.198$ $p = 0.906$
likability	0[0,1]	0[0,1]	0[0,1]	$\chi^2 = 1.339$ $p = 0.512$

If one looks closer into the friendliness values (see Figure B.1), test condition *happiness expression* appears to have higher values than test condition *neutral expression*. In turn, test condition *neutral expression* presented values for friendliness higher than test condition *sadness expression*. On the whole, the data does not contradict our idea that face expressions alone do not affect the perception of the analyzed features, except for friendliness. Therefore the results presented for friendliness should be faced with skepticism. Hence we do not mention this feature in the Evaluation's concluding remarks. Note however that likability was not significantly affected, hence the results referring to it remain valid.

These tests may have partially validated the Evaluation's results, nonetheless the manipulation could have potentially created a new problem: the pre-test no longer being a valid baseline due to a carry-over effect. To probe this potential problem we did a comparative analysis between test conditions of feature values in section "Act 1 Questions". We used a series of Kruskal-Wallis tests whose results are presented on Table B.4.

As can be seen in the results, the test conditions did not have significantly different ($p \geq 0.05$) feature values, hence there was no effective carry-over effect from the "Introduction" manipulation. This might be the result of participants having been exposed to more media concerning meemos (some text and a video) when they answered the questions in "Act 1 Questions". The

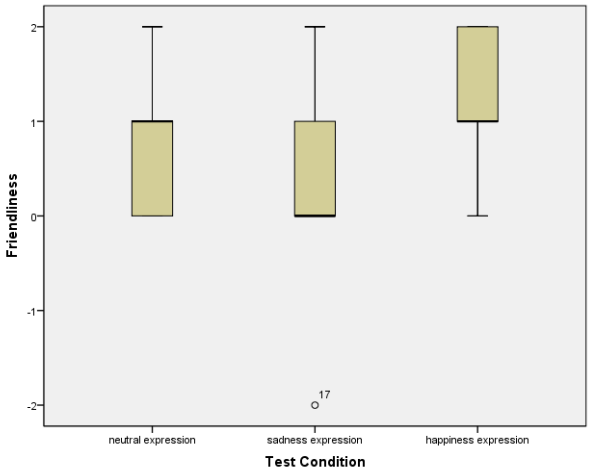


Figure B.1: Box plot for friendliness in "Expressions Questions"

effect of the initial image had "blurred" by then.

B.4 Summary

When analyzing emotion identifiability, we achieved recognition rates between 74% and 97%. We found these values to be adequate, as they were similar to the ones obtained by Canamero [9] in a comparable experiment. We proceeded by considering what emotional reaction did participants expect in "Act 2". A percentage of 66% claimed expecting an emotional reaction, 60% of which said that this reaction could be *sadness*. Nonetheless *fear* was also selected by many (40%) which might hint that an architecture supporting episodic memory retrieval should also support expectation related emotions. Lastly, we considered an additional manipulation used to validate the Evaluation's results. We concluded that the friendliness results were significantly affected by the facial expressions alone, hence should be left out. Then we tested for a possible carry-over effect of this manipulation, finding it non-significant.

Table B.4: Kruskal-Wallis for features in "Act 1 Questions" (2 degrees of freedom)

features	Descriptive Statistics			Kruskal-Wallis differences between conditions
	neutral expression (N=31) Mdn[Quartiles]	*sadness expression* (N=30) Mdn[Quartiles]	*happiness expression* (N=29) Mdn[Quartiles]	
awareness	0[-2,1]	0[-0.25,1]	0[-1,1]	$\chi^2 = 0.925$ $p = 0.630$
behavior understandability	1[0,1]	1[0,1]	1[0,1]	$\chi^2 = 0.256$ $p = 0.880$
personality	1[0.75,1]	1[0,1]	1[-0.5,1]	$\chi^2 = 0.991$ $p = 0.609$
visual impact	0[-0.25,1]	1[0,1]	0[-1,1]	$\chi^2 = 3.607$ $p = 0.165$
friendliness	1[1,1]	1[0,2]	1[0.5,2]	$\chi^2 = 2.029$ $p = 0.363$
intelligence	0[-1,1]	0[-1,0]	0[-1,1]	$\chi^2 = 0.180$ $p = 0.914$
likability	0[0,1]	1[0,2]	0[0,1]	$\chi^2 = 1.291$ $p = 0.524$
interest	0[-1,0.25]	0[-1,1]	0[-1,0]	$\chi^2 = 1.530$ $p = 0.465$