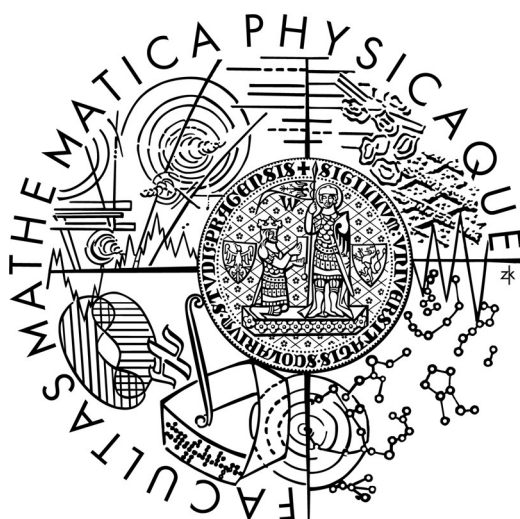Charles University in Prague
Faculty of Mathematics and Physics

MASTER THESIS

Bc. Michal Bída
Emotional virtual agents
Department of Software and Computer Science Education
Supervisor: Mgr. Cyril Brom, Ph.D.
Study program: Informatics, Theoretical Computer Science (Artificial Intelligence)

Acknowledgments

I would like to thank Pogamut team for their support when I was developing project Emohawk and Tomáš Plch for his feedback. Also I would like to thank my supervisor Mgr. Cyril Brom, Ph.D. for leading this master thesis and providing the feedback.

I hereby declare, that I wrote the master thesis autonomously using exclusively the sources listed in the bibliography. I approve its lending.

In Prague at 6 August 2009   *Michal Bída*

Always listen to experts. They'll tell you what can't be done and why. Then do it.

(Robert A. Heinlein)

Table of contents

Název práce: Emoční virtuální agenti
Autor: Michal Bída
Katedra (ústav): Kabinet software a výuky informatiky
Vedoucí diplomové práce: Mgr. Cyril Brom, Ph.D.
e-mail vedoucího: *brom@ksvi.mff.cuni.cz*

Abstrakt: Vývoj inteligentních virtuálních agentů (IVA) je náročný proces obsahující mnoho dílčích podproblémů. Teoretická výuka se v této oblasti může opřít o několik vynikajících publikací. Naproti tomu praktickou výuku je v této oblasti možné provádět jen omezeně. Několik málo platformem, které se tímto zabývají, mnohdy nesplňuje všechny požadavky, které jsou na ně kladeny. Cílem naší práce je vytvořit platformu umožňující výuku v oblasti vývoje IVA. Prvním krokem k tomuto cíli je vytvoření prototypu implementace této platformy – projektu Emocuc. Projekt Emocuc bude obsahovat IVA ovládané architekturou, která bude využívat afekty získané pomocí psychologicky plausibilního emočního modelu. Tito IVA budou schopni sehrát krátkou scénku s nelineárním (emergentním) příběhem. Součástí práce bude vytvoření metodiky, která umožní analyzovat uvěřitelnost a emergenci v projektu Emocuc.

Klíčová slova: inteligentní virtuální agenti, výuka, emoční model, architektura využívající afekty

Title: Emotional virtual agents
Author: Michal Bída
Department: Department of Software and Computer Science Education
Supervisor: Mgr. Cyril Brom, Ph.D.
Supervisor's e-mail address: *brom@ksvi.mff.cuni.cz*

Abstract: The development of intelligent virtual agents (IVAs) is a complex task featuring many sub problems. Concerning the education in this field, there is a good theoretical basis. However when it comes to the practical education – the platforms that can be used are scarce and mostly still not fully developed. Our goal is to create a platform, which would allow for a good practical education in the field of IVAs development. The first step towards this platform is a prototype implementation – project Emohawk – that will be described in this thesis. Project Emohawk features a partly emergent story and affect-driven architecture for IVAs control based on a psychologically plausible emotion model. Moreover a methodology was created analyzing this prototype implementation regarding the believability and emergent story potential.

Keywords: intelligent virtual agents, education, emotion model, affect-driven architecture.

1 Introduction

With computers getting faster there is a growing potential in simulations we can perform on them. Nowadays the virtual environments in computer games or other applications look amazingly real. And so does the graphical representation of the virtual inhabitants of these environments – the intelligent virtual agents (IVAs). The goals of these virtual inhabitants differ according to the environment they live in. Sometimes they live a peaceful family life (e.g. [EA, 2009]). In other applications they are violent beings trying to harm others [Epic, 2004]. Nevertheless both kinds of these IVAs have their goals, can perform certain actions in the environment, produce simple *stories* of their life and are trying to be *believable*. What does it mean, "are trying to be *believable*"?

If we put it simply a *believable* IVA is one that successfully convinces the user that it is an autonomous being with own goals and motives in the environment he is living in. More accurate definition of believability along with aspects IVAs should satisfy in order to be believable is provided in [Loyall, 1997].

Concerning the *stories* of IVAs life we can tell for example that the computer games feature high number of different stories with many plots and twists. The characters are falling in love, the worlds are being saved and the kingdoms are being destroyed. However due to rapid and effective development of computer games these stories tend to be scripted. The user usually cannot affect the story much. Sometimes he can decide to be good or evil. Sometimes he can decide not to complete some side story. But this is all. In modern computer games the focus is mainly on computer graphics. The other aspects of game development need to be effective as much as it is possible. Due to these reasons the developers are often using scripted stories to save the development time. In IVA research community the *virtual storytelling* is concerned with story generation for applications featuring virtual worlds.

Let's imagine our own dream game for a while. It would feature a virtual town inhabited by many IVAs. Each of these IVAs would pursue own goals and could interact with other IVAs. We would have policemen, teachers, vendors and many other IVAs in this town. Suddenly a murder would take place. At this time human user controlling one IVA would be summoned to the town with a goal of finding out who was the murderer. How he will start? Will he go to the police station and ask details about the crime? Will he try to find some witness who could see something important? Every user decision and interaction would have effect onto the story outcome. For example the interrogation of witnesses before going to the police station could have a negative effect onto policemen willingness to tell details about the crime. Or collecting enough evidence against the murderer could drive the murderer to flee from town. However none of these behaviors would be scripted. These behaviors would be a result of user – agent interactions and agents internal states (e.g. Is the murderer scared enough to flee? Is a policeman angry enough not to respond to user's questions? Or will the policeman express his anger otherwise?). The outcome story would emerge from the basic user – agent interactions and would be shaped by every user decision.

If we would develop a system capable of producing the story mentioned above in a believable way we would develop an *emergent narrative* system. Emergent narrative development is recently attracting many IVA researchers (see Chapter 2). The definition of emergent narrative (provided by [HUMAINE wiki, 2009]) is:

"A participative style of narrative in which the lower-levels of narrative structure emerge from interaction between characters rather than being scripted as part of a pre-defined plot."

The development of emergent narrative is discussed in greater detail in [Louchart et al., 2008]. What advantages emergent narrative brings? The computer games featuring non-linear stories could be more entertaining for the users. The serious games (games developed for education) featuring open worlds where anything can happen would be more believable, hence offering better education basis.

Let's go back to IVAs. We want them to be believable and be able to produce emergent stories. Both of these goals are not easy to achieve. The developer requires many skills and knowledge to effectively solve all the problems associated with these issues. What problems are we talking about?

Firstly, concerned with the believability of IVAs, we may require some kind of emotion modeling in order to simulate human being in a believable way. Creation of a psychologically plausible emotion model is a challenging task on its own. Moreover the integration of an emotion model within the IVA decision making is necessary.

Secondly, the design of emergent stories presents an hard to tackle issue, where a system would: a) allow for open ended stories shaped by the user interaction and b) be believable at the same time (e.g. not containing "dead ends" in the story, be able to deal with all possible user interactions etc.)

These problems are indeed complicated. Education in this field is necessary in order to help students and newcomers overcome them. The theoretical knowledge in this field is covered by many great books such as [Rabin, 2002-2008] or [Champandard, 2003]. However, the field of practical education on this topic lacks the necessary tools. To our knowledge there are only three tools that support education explicitly – Netlogo, Alice and Pogamut. Netlogo [Wilensky, 1999] is a great tool for building simple agents and running social simulations. Unfortunately it lacks a complex 3D environment, hence it is not suitable as a main tool for IVAs development education. Alice [Alice, 1995 – 2009] is a tool to learn programming by means of agents. Although Alice offers better support for general education in programming, its built-in 3D environment is not comparable to current game engines making it not a platform of choice. Pogamut [Kadlec et al., 2007] is a tool based on Unreal Tournament 2004 (UT04) [Epic, 2004] developed by AMIS group [AMIS] at Charles University in Prague. It already has been successfully used in undergraduate as well as high-school education. Pogamut is primarily oriented on gaming AI audience, lacks non-violent graphic and has limited support for non-gaming AI issues.

None of presented tools support fully IVA development education. Our goal is to overcome this gap by developing educational toolkit Emohawk facilitating the teaching of:

1) IVAs basic control in 3D world, e.g. steering mechanisms and reactive planning,

2) emotion modeling and IVA believability issues,

3) multiple IVAs coordination in the means of story directing,

4) creation of a partly emergent story,

5) basics of 3D animation, e.g. generation of walking behavior and gestures.

This toolkit should be built upon Pogamut (the newest version – Pogamut 3). We have chosen Pogamut because of its extensibility and our previous working experience with it. The Emohawk toolkit should be an interactive application featuring partly emergent stories and taking place in 3D virtual world (IVA research community would describe the Emohawk toolkit as an interactive storytelling application). The user will be allowed to interact with IVAs either directly by participating in the story or indirectly by changing the IVAs internal

states or by modifying the environment (e.g. adding some items to the environment, etc.). IVAs should be believable and controlled by affect-driven architecture (decision making architecture exploiting affects from some emotion model). More details about our Emohawk toolkit vision can be found in [Brom et al., 2009].

To summarize we want to create an application featuring emergent story and believable IVAs controlled by architecture exploiting affects. We already know there are several problems connected with this. These problems especially are:

a) **Affect-driven architecture.** Architecture exploiting affects requires psychologically plausible emotion model to be a part of the decision making system. This does not only regard the computational part that provides us with actual emotions, but also the expressive part that somehow visualizes computed emotions. Emotion models provide the computational part. The expressive part (how we will use and express the emotions) will be implemented by us. We will also need to implement some kind of mapping of events in the environment to emotion model input. We will need to solve issues as which events in the environment should trigger changes in the emotions, etc. There are indeed many challenges connected with these goals – especially believability issues and model balancing issues.

b) **Emergent story implementation.** The emergence in our project should come from basic interaction between agents that will be shaped by affects and random components of our platforms – implemented by us or natively present in the used platforms. How will we define these interactions and their impact on IVAs decision making? And when we are done how will we evaluate that the stories produced are believable? Since the story is emergent there could be many branches we would need to explore to see all the possible outcomes? This leads us to the last problem below.

c) **Scenario analysis.** When our implementation is done, we will need to evaluate it in the means of believability and emergent story potential. We will need to answer questions as: Are our IVAs believable? How many possible story outcomes are present in our scenario? And more. To take a grasp of this, we will need to run the scenario multiple times (even hundreds of times). Since it is not possible to watch every run of the scenario, we need some methodology how to analyze this at least partly automatically. The creation of such methodology is one of the challenges we will face.

The outcome of this thesis is a prototype implementation of the educational Emohawk toolkit. We will refer to this prototype as to project Emohawk. The purpose of this prototype implementation was to pinpoint potential design and architecture problems as well as to create affect-driven architecture and confirm it produces acceptable behavior. The second goal was to create and use a methodology with which we can evaluate project Emohawks performance in the means of producing believable IVAs behavior and believable, partly emergent story.

The project Emohawk is based on Pogamut 3 platform (its beta implementation). As a virtual world it utilizes the action game Unreal Tournament 2004 (UT04). Project Emohawk IVAs are written in Java and their affect-driven architecture is using the emotion model ALMA to provide affects.

Note that although many related work projects (presented in Chapter 2) use planning for controlling the emergent story we will not use planning in project Emohawk. The reasons are two: a) our project will feature just one simple story (presented below) and b) we will be concerned more with agent to agent interaction than with user to agent interaction (this will

give us more control over the story). More stories based on our scenario with better story control will concern us in future work.

Project Emohawk scenario features currently one simple story. This story was created by a professional female writer according to the following conditions:

1) scenario should be interesting for both boys and girls

2) scenario should feature a story (at least a simple one)

3) scenario should be relatively short (~ 10 minutes)

4) scenario should be interactive and spectator-based at the same time

5) scenario characters should exhibit emotions

6) scenario should be close to real world – we allow there a few non realistic things, but sci-fi bases or medieval knights are not allowed

7) there should be some kind of interaction between characters, that would be reasonably simple and believable at the same time

The current scenario story setting (description based on [Brom et al., 2009]) is as follows:

There is a small city with about a thousand of inhabitants. There lives Bruno, age 19, in the town. Bruno has a girlfriend Anne and... well... yet another girlfriend Clementine. The girls do not know about each other.
Bruno has just been with Anne in the cinema. Now, he has to walk her home quickly since Clementine is waiting for him to be taken to the very next movie. Besides, there is an emohawk roaming around. Emohawk is an alien creature that is found adorable by all IVAs in the story.

The interaction between characters is handled symbolically through actions, proposals and casual conversation. The story outcome as well as the exact shape of the story is emergent although there are some patterns the story follows. In the chapter below we will summarize goals and outcomes of this work.

1.1   Goals and outcomes

In this chapter we will summarize project Emohawk goals and outcomes. There were three main goals of project Emohawk:

1) Implement affect-driven architecture for IVAs control.

2) Use these IVAs to create an example scenario featuring a simple, partly emergent story.

3) Create a methodology that can be used to analyze the resulting scenario performance in the means of believability and emergence.

Since the project Emohawk was based on multiple platforms (Pogamut 3, ALMA and UT04), the implementation was divided among them. Below we will provide a summary of all the outcomes of project Emohawk. We can split the work on our project into four categories.

1) Implementation and testing of the new Pogamut 3 core. Pogamut 3 core is still in development. We have helped to debug Pogamut 3 core and extended it to suit project

Emohawk needs.

2) Extending UT04 content for project Emohawk. UT04 lacks non-violent models. Also we needed to perform several other modifications to UT04.

3) Implementation of project Emohawk in Pogamut 3 (programming IVAs decision making, connecting emotion model ALMA to Pogamut, etc.)

4) Creating and using a methodology to analyze the project Emohawk scenarios.

Especially in the scope of this thesis following things were accomplished:

1) Extension of UT04 that added new sensors and actions, civil (non-violent) models for IVAs and new emotion display capabilities.

2) Testing and debugging new Pogamut 3 core.

3) Connecting emotion model ALMA to Pogamut 3.

4) Implementation of agents affect-driven control architecture.

5) Implementation of the project Emohawk example scenario.

6) Creation of a methodology enabling to partially automatically analyze the outcome scenario.

7) Using this methodology to evaluate project Emohawk performance.

Our evaluation showed that our example scenario produces emergent story outcome with reasonable level of believability for the project in an early stage. However there are some problems as well. Below we will present the reader with outline of this work.

1.2 Outline

This work is organized as follows. In Chapter 2 we will present related projects. In Chapter 3 we will describe platforms and tools we have used to implement project Emohawk. Chapter 4 will provide a brief introduction into emotion modeling issues and presents emotion model ALMA. In Chapter 5 we will describe project Emohawk in a greater detail. Chapter 6 will be dedicated to implementation, Chapter 7 will be dedicated to evaluation of the project. Chapter 8 will follow with discussion and future works. After that we will conclude this thesis.

2 Related work

Here we will present the reader with related work in the field of emergent narrative. We will consider three projects – Facade, Madame Bovary and FearNot! and provide brief comparisons with our project in the summary section of this chapter.

2.1 Facade

In this section we will present project Facade. Description here is based on paper [Mateas & Stern, 2003]. Facade [Mateas & Stern, 2003] is an interactive drama finished in 2005 by Michael Mateas and Andrew Stern. In Facade the player takes role of a long-time friend of Trip and Grace – a married couple in their thirties. The player is invited to their apartment for the evening that quickly turns ugly because of Trips and Graces marriage problems. How will the story unfold as well as the final outcome of the story, that depends on the player interaction – his actions and conversation during the drama.

Fig. 1. Interactive drama Facade. On the figure we can see Facade IVAs – Trip and Grace.

Technically speaking the Facade is a first-person non-violent 3D game. The environment consists primarily of large furnished living room with animated agents Trip and Grace (Fig. 1). The movement is controlled by arrow keys and the player can communicate with Trip and Grace by typing the text input or by doing one of three predefined actions – kiss, hug or comfort (these are triggered by a mouse). For programming the characters and the

Facade story, number of authoring languages was developed. We would like to mention "a behavior language" (ABL), which is a reactive planning language based on Hap [Loyall & Bates, 1991] and "beat sequencing language", which is a language used for drama management.

The story coherence is maintained and controlled by drama manager (DM). This DM continuously monitors the simulation and regularly updates rules of the simulation according which Trip and Grace operates. These updates are organized into so called story beats. Each beat is a collection of behaviors tailored to a particular situation or context. According to [McKee, 1997] beat is the smallest unit of dramatic action that moves the story forward. Each beat has its preconditions and effects on the story state specified by authors. DM uses this information to create an overall dramatic narrative – a plot. The preconditions and effects of the beats specify in fact a partial ordering. The Facade uses a pool of ~200 beats with one beat being active for approximately one minute. With average duration of one story being something between 10 – 15 minutes at most, the player will not experience more than ~18 beats per one simulation. The Facade beat approach allows the story to be non-linear and coherent, but with the cost of time consuming authoring of all of the content. The beat approach is being suitable for tight scenarios, where every player action has some effect on the story.

2.2 Madame Bovary

In this section we will present project Madame Bovary. The description here is based on paper [Cavazza et al., 2007]. Madame Bovary or Madame Bovary on the "holodeck" [Cavazza et al., 2007] is a real-time immersive interactive storytelling system. In this system the user has the possibility to influence unfolding of the story by his action. The story is based on the classic novel by Gustav Flaubert – Madame Bovary. The system is capable of real-time stereoscopic visualization of used Unreal Tournament environment. The characters express themselves using speech and body gestures. The user interacts with the characters by gestures and natural language.

To control the story unfolding, planning mechanism was used. The virtual characters are equipped by a set of feelings. These feelings serve as goals of the planner. Each character is equipped by a planner using Heuristic Search Planning (HSP) in real-time implementation. For the storytelling engine Madame Bovary uses character based approach. Each character is driven by his/her own set of feelings. The interaction of characters uses synchronization – waiting for answer state. Some of the feelings used in the scenario are:

– anger_towards(x, y)

– accepts_adultery_risks(x, y)

– power_over(x, y)

The "x" represents the target character here, while "y" the intensity, which may be low, medium or high.

The authors treat Madame Bovary as a proof-of-concept work in the field of fully immersive emergent interactive narrative. This is system is currently able to produce stories of duration of 2 minutes, although not fully immersive desktop version produces stories up to 6 minutes of duration.

2.3 FearNot!

In this section we will present project FearNot!. Description here is based on [Aylett et al., 2005]. FearNot! (Fun with Empathic Agents to Achieve Novel Outcomes in Teaching) [Aylett et al., 2005] later redesigned according to evaluation [Hall et al., 2006] is an application of virtual drama for anti-bullying education of 9 to 13 aged children. In FearNot! the user (usually a child) takes a role of an invisible friend of a victim of bullying. The victim is a small children boy or girl aged 9 to 13 that is a newcomer to a local school. The victim experiences continuous bullying by one of the classmates. After each bullying episode the victim asks the user for advice in dealing with this. The user can suggest a coping behavior, which may be or may be not used by the victim in the next bullying episodes. Whether the coping behavior will be selected or not depends on the nature of next bullying episode as well as on the victim's current emotional state.

Fig. 2. FearNot! 3D environment. On the figure we can see FearNot! 3D environment with two FearNot! IVAs – victim (lying on the ground) and the bully (standing next to victim).

FearNot! features 3D world with 3D rendered characters that are able to speak (Fig. 2). The user is a spectator and cannot affect the story except the after bullying sessions with the victim. There the user can communicate with the victim through text input and can affect the victims decision making in the next bullying episodes. The story unfolding in FearNot! is emergent (not scripted) and controlled by the stage manager [Sobral et al., 2003]. The affect-driven agent architecture controlling the characters further promotes the emergent narrative mechanism.

The stage manager is the key component of FearNot! story control. It contains the conditions capturing the developer's needs and a set of episodes. The developer's needs define a set of constraints that must be satisfied during the episode and during the whole story. For example – there must occur at least one action every x seconds in the episode or it will end. The episodes define additional context for the characters. For example an episode may be: John (the victim) is in the student's canteen dining with other children, when suddenly Luke (the bully) comes in front of him. This episode would qualify as a conflict episode, but would not guarantee the actual occurrence of a conflict. This depends on the characters decision making, which is defined in a way the conflict will most likely occur. The bully has the goal to bully the others, while the victim is programmed to react to those events. Furthermore each episode is annotated with a set of preconditions, which should be fulfilled before the episode is actually selected by the stage manager. This with a set of expected post-conditions after the episode ends helps the stage manager to decide which episode should be selected next.

Now the character control architecture has also a big impact on the episode performance. Every character has got a set of goals he is pursuing – a bully to bully others, a victim to stay unharmed etc. Also the characters feature an emotional model inspired by OCC theory. The emotions have huge impact onto characters as they control decision making by making some goals more or less desirable. Moreover the emotions may even directly trigger certain actions – e.g. character will cry if the situation will be too much unpleasant. The way the characters are programmed assures there will usually be conflicts between the bully and the victim, although the exact shape of the conflicts is emergent. In the same situation the bully may insult the victim verbally or even physically attack the victim or do something else.

If we summarize the approach the FearNot! emergent narrative comes from combination of careful predefined setting of the scene by the stage manager and the agent affective architecture that controls the characters in partly anticipated yet still emergent manner.

2.4 Summary

We have presented three projects concerned with emergent narrative. The Facade's approach allows a good balance between story control and user interaction affecting the story. However this is achieved by extensive authoring work (which we cannot afford due to higher number of planned scenarios). Also Facade beat approach suits better tight, closed scenarios. In our Emohawk educational toolkit we plan to implement rather higher number of different scenarios. These scenarios will take place in open world not as limited as Facade's environment. Concerning this vision there are other ways more effective than the beat approach.

Madame Bovary goal is to provide fully immersive storytelling environment. This rather differs from our goal, as we require the immersion not. The Madame Bovary storytelling engine is based on planning and agent affects – feelings – that serve as goals for the planner. Our affect-driven architecture will use simple feeling as well, although in a different way, due to the absence of any planner in project Emohawk. At the moment our story is simple enough, we are able to generate it just with reactive behavior. Also the user – scenario interaction is currently just a minor goal. Hence we will not incorporate planning at this time. Since we do not plan to incorporate planning at this time and the planning is the key component of Madame Bovary decision making system, it makes Madame Bovary approach rather different from ours.

FearNot! is a great experiment in emergent narrative. However it features somewhat

tight scenario setting, without user interaction we will be concerned with later on (user can only give advices, he cannot perform actions in the virtual world). Still, there are some commonalities of our two projects. As in FearNot! the emergence in project Emohawk should come from agent to agent interaction, which is shaped by affects. The scenario setting in project Emohawk is designed in a way that promotes occurrences of conflicts as in FearNot!. Unlike FearNot! our project will feature just one scenario setting at the movement requiring not any planner as in FearNot!.

Planning is one of the approaches that is widely used by projects concerned with emergent narrative, but will be missing in our current implementation.

The biggest advantage of our project concerning the related work here will be the a) unlike some of the projects described here our scenario is open source and b) the environment of UT04 is extensible and rich making future extensions of our scenario easy and possible.

3 Used platforms

Project Emohawk is based on the Pogamut 3 platform [Pogamut, 2009]. Pogamut is an IDE for IVAs development. As its virtual world the platform uses first-person action game Unreal Tournament 2004 (UT04). To connect to the virtual world of UT04 project GameBots was used. To model project Emohawks IVAs affects we used emotion model ALMA. Below we will describe all mentioned tools and platforms except ALMA in a greater detail. ALMA emotion model will be described in a following chapter.

3.1 Unreal Tournament 2004

Unreal Tournament 2004 (UT04, Fig. 3) from [Epic, 2004] is a first-person action game. It provides us with simulated 3D environment with a vast amount of different locations ranging from sci-fi space stations or old factories to forests, meadows and small towns. Big advantage of UT04 is that it features its own scripting language UnrealScript, which is used to program the game mechanics. Everything except the 3D engine is open source and can be modified by the user. This makes the game easily modifiable. Also there is huge on-line community around this game providing us with new items, locations or game modifications. Part of the game is an integrated editor UnrealEd, which makes modification of the game content even more accessible.

Fig. 3. Unreal Tournament 2004 environment. Two project Emohawk IVAs in front of the park. Around their heads we can see colored flares used to express IVAs current emotions. Copyright Epic
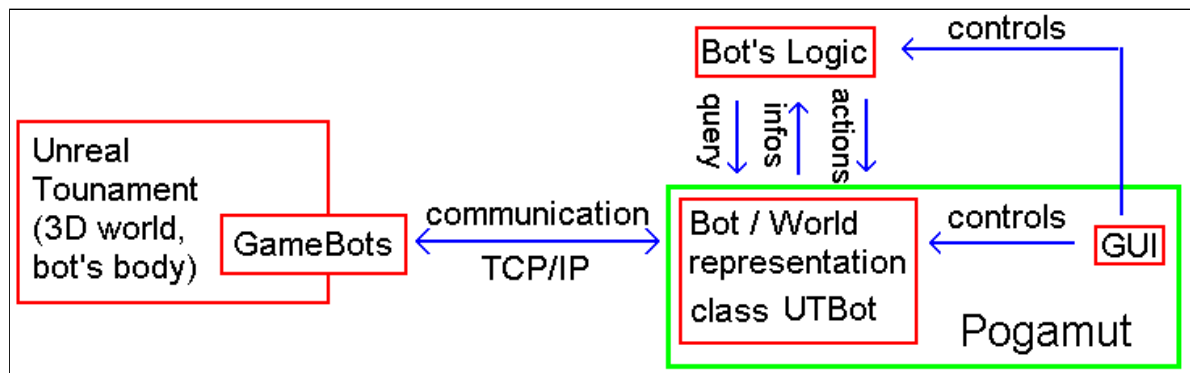
3.2 GameBots

GameBots originally created by [Adobbati et al., 2001] are a modification (mod) for UT04. Their main purpose is to make rich environment of UT04 available for virtual agents development. GameBots provide network TCP/IP text protocol for getting information about environment and controlling in-game avatars. GameBots are programmed in UnrealScript scripting language – programming language that was created and used to program UT04 game mechanics (almost everything except the graphical engine). UnrealScript resembles Java or C programming language.

Considering project Emohawk purposes we have modified GameBots to better suit project Emohawk needs.

3.3 Pogamut

Pogamut is a tool for the IVAs development. Our Emohawk project used newest Pogamut version (Pogamut 3) [Pogamut, 2009]. The Pogamut 3 is a work in progress. At the time of writing the base Java library was ready and stable and the implementation of platforms IDE was partially complete. Pogamut 3 architecture is shown on Fig. 4. Pogamut features sensory methods that can be used to query agents state and as well effector methods for controlling the agent in the environment. Pogamut 3 virtual environment is Unreal Tournament 2004.

Fig. 4. Architecture of Pogamut. GameBots are used to connect to UT04 through text protocol. NetBeans Java Client handles the connections to the environment and provides our agent with sensors and actors methods (e.g. seeing item, go to location, etc.). GUI can be used to manage the agents and to debug their decision making systems.

Thanks to Pogamut platform we have an easy access to UT04 for project Emohawk purposes.

4   Emotions and ALMA – A layered model of affect

Before we venture in details about emotion model ALMA, we would like to provide a brief introduction to problems of emotion modeling for IVAs. Human emotions are a multi-faced theme. There are many different approaches how to study emotions, focusing on different phenomens the emotions feature [Plutchik, 2002]. Also the role of emotions in virtual agents is not clear-cut. Emotions can affect decision making [Bida & Brom, 2007], expressive part of agents' behavior [Bevacqua et al., 2007a] or just serve the agent as another source of information when interacting with humans [Bevacqua et al., 2007b]. Often the emotions are used as a way to increase IVAs believability due to the fact, the emotions play an important role in human behavior [Damasio, 1994].

To provide IVAs with emotions, emotion models are being developed. There are strong requirements on these models. Their implementation should provide a) psychologically plausible way of modeling the human affect and b) clear, standardized ways of input and output. A creation of such a model is not an easy task, requiring knowledge from multiple fields.

To provide project Emohawk IVAs with affects, we have decided to use emotional model ALMA. ALMA's big advantages are a) this model is universal and not bound to any environment, b) this model is written in Java – the same language as Pogamut (this makes ALMA connection to Pogamut easy), c) this model features several, distinct affect types, d) this model is free for non-commercial purposes and e) one of the concerns of this model was psychological plausibility.

In the rest of this chapter we will introduce emotion model ALMA – A layered model of affect [Gebhard, 2005] that is based on OCC cognitive model of emotions [Ortony et al., 1988]. ALMA is a Java engine with GUI that can provide virtual agents with affects. ALMA features three types of affect that differs in the means of duration. It provides us with emotions for short-term affect, moods for medium-term affect and personality for long-term affect. With ALMA our virtual agents will be capable of emotional reactions to events in the environment (we will know for example what made our virtual agent to be angry). However ALMA does not specify how these emotional reactions should be used to alter the agent's behavior. Also the mapping of environmental events to ALMA inputs is pretty much in the hands of the developer. This enables on the one hand universal use of this model, on the other hand the user will always need to provide a mapping from environment to ALMA input as well as to create the actual changes in IVAs behavior according to ALMA affects (which would one expect). The input and output features of ALMA as well as ALMA GUI will be discussed below.

4.1   ALMA input

The form of ALMA inputs is inspired by OCC model. Hence we will briefly present the OCC theory of emotions here.

OCC theory. Emotional theory of Ortony, Clore and Collins was designed in 1988 especially for the purpose of implementation in the computer systems. The theory makes a presumption that emotions are a result of certain sequences of cognitive appraisals and interpretations. Thus it is a cognitive theory of emotions.

According to authors the emotions are valenced affective reactions. Their purpose is to evaluate events that happen in the world. There are three main groups of emotions in the OCC

model: emotions as reactions to events, emotions as reactions to agents and emotions as reactions to objects (agents are meant as living while objects as inanimate). Furthermore the theory specifies a set of emotional variables that are used to appraise the events in the environment. These variables can be either global (affecting all the emotions) or local (affecting just certain emotion category). To get the information about which emotions should be associated with certain event, agent or object in the environment, these variables are being processed with a set of if-then rules. The result is then a set of emotions associated to certain event, agent or object.

Most important variables in the model are:

– *desirability* measuring the outcome of an event (1 to -1, pleasant or unpleasant) and affecting emotions attributed to events.

– *praiseworthiness* measuring our attitude towards action of different agent (-1 to 1, I approve or disapprove the action) and affecting emotions attributed to agent .

– *appealingness* measuring our personal attitudes toward objects (I like or dislike the object) and affecting emotions attributed to objects.

The process of generating emotions in OCC model can be generalized as follows: Some event occurs in the environment. After this the event is appraised by a set of OCC variables. These variables are passed to if-then rules. This results in a set of OCC emotions associated with the event (and/or agent and/or object).

ALMA. Input to ALMA is provided through emotion eliciting conditions (EECs) either directly by setting the emotion eliciting conditions variables or by using some pre-defined substitute method. The emotion eliciting conditions variables are a sub set of OCC theory variables. List of EECs variables (ranging from -1 to 1) follows:
– *desirability* – used to measure an event outcome, if desirable (in accordance with agents goals) the desirability will be set to positive and vice versa
– *praiseworthiness* – used to measure agents actions, if we approve the action we will set this to positive value and vice versa
– *appealingness* – how much we like object or agent that is taking a part in recent event
– *likelihood* – used to measure prospect based action – how likely the event will happen
– *liking* – how much the agent is attracted to another person
– *realization* – some prospect based event finally occurred – how much it was realized.
– *Agency* – can be set to "self" or "other" providing information about whether this event or action is caused by us or by some other agent
– *elicitor* – who caused this event or action. May be some other agent or some other conditions.

The filled EEC is then processed by a set of ALMA emotion generating rules and a set of OCC emotions is generated. The intensity of generated emotions is also affected by the agent's personality. The emotions are then used to shift the mood of the agent. The emotions in ALMA are decaying over time.

Summary. The input to ALMA is provided by emotion eliciting conditions. According to their variables and agents personality a set of OCC emotions is generated and these emotions are then used to affect mood.

4.2 ALMA output

As noted above ALMA provides three distinct affect types: emotions as short-term affect, mood as medium-term-affect, and personality as long-term affect.

Emotions. Emotions in ALMA reflect short-term affect. They are usually bound to certain event, action or object, which is the cause of this emotion. The emotions in ALMA are decaying over time until they completely disappear of the agents focus. In ALMA agent can have multiple emotions of the same type active at one time, but always the emotion with the biggest intensity among the emotion type is in his focus. ALMA features 24 OCC emotions.

The complete list of emotions follows: admiration, anger, disliking, disappointment, distress, fear, fearsConfirmed, gloating, gratification, gratitude, happyFor, hate, hope, joy, liking, love, pity, pride, relief, remorse, reproach, resentment, satisfaction and shame.

Mood. Mood in ALMA is based on [Mehrabian, 1996] mood model. The mood in ALMA is understood as a medium-term affect, which is generally not related with concrete event, action or object. Mood should be longer lasting affect, more stable than emotions.

ALMA mood is represented by three dimensions – pleasure (P), arousal (A) and dominance (D). These dimensions are nearly independent of each other and form three dimensional PAD space. According to octen the mood is currently in, the ALMA recognizes 8 moods shown in Table 1 below.

+P, +A, +D	Exuberant	-P, -A, -D	Bored
+P, +A, -D	Dependent	-P, -A, +D	Disdainful
+P, -A, +D	Relaxed	-P, +A, -D	Anxious
+P, -A, -D	Docile	-P, +A, +D	Hostile

Table 1. ALMA moods. Each cell shows the needed dimensions value (positive "+" or negative "-") for the mood in following cell to occur. Table based on mood table from [Gebhard, 2005].

The mood in ALMA is computed through agent current emotions. Each emotion is mapped onto the 3D PAD space (Fig. 5). Then the average of all emotions in the PAD space is computed and the mood is attracted to this spot. Moreover, when the mood reaches this spot, second phase of the mood computation mechanism will be triggered and the mood will be continuously reinforced in the direction from neutral mood (0,0,0) to the current emotion average.

Emotion	P	A	D	Mood Octant
Admiration	0.5	0.3	-0.2	+P+A-D Dependent
Anger	-0.51	0.59	0.25	-P+A+D Hostile
Disliking	-0.4	0.2	0.1	-P+A+D Hostile
Disappointment	-0.3	0.1	-0.4	-P+A-D Anxious
Distress	-0.4	-0.2	-0.5	-P-A-D Bored
Fear	-0.64	0.60	-0.43	-P+A-D Anxious
FearsConfirmed	-0.5	-0.3	-0.7	-P-A-D Bored
Gloating	0.3	-0.3	-0.1	+P-A-D Docile
Gratification	0.6	0.5	0.4	+P+A+D Exuberant
Gratitude	0.4	0.2	-0.3	+P+A-D Dependent
HappyFor	0.4	0.2	0.2	+P+A+D Exuberant
Hate	-0.6	0.6	0.3	-P+A+D Hostile
Hope	0.2	0.2	-0.1	+P+A-D Dependent
Joy	0.4	0.2	0.1	+P+A+D Exuberant
Liking	0.40	0.16	-0.24	+P+A-D Dependent
Love	0.3	0.1	0.2	+P+A+D Exuberant
Pity	-0.4	-0.2	-0.5	-P-A-D Bored
Pride	0.4	0.3	0.3	+P+A+D Exuberant
Relief	0.2	-0.3	0.4	+P-A+D Relaxed
Remorse	-0.3	0.1	-0.6	-P+A-D Anxious
Reproach	-0.3	-0.1	0.4	-P-A+D Disdainful
Resentment	-0.2	-0.3	-0.2	-P-A-D Bored
Satisfaction	0.3	-0.2	0.4	+P-A+D Relaxed
Shame	-0.3	0.1	-0.6	-P+A-D Anxious

Fig. 5. Mapping of OCC emotions into PAD space. Figure taken from [Gebhard, 2005]. P stands for pleasure, A for arousal and D for dominance.

Personality. Personality in ALMA uses big five model described in [McCrae & John, 1992] and reflects ALMA's long-term affect. User sets it up for each agent before the ALMA starts and it does not change during run-time. It affects OCC emotion intensities as well as the agents default mood. The five dimensions of big five personality model are as follows:

– *openness* – reflects interest in intellectual issues, unconventional values, aesthetic sensitivity, need for variety.
– *conscientiousness* – reflects task-oriented characteristics such as being dependable, responsible and orderly.
– *extraversion* – reflects a tendency to be sociable and experience positive affect.
– *agreeableness* – reflects a tendency to be inter-personally pleasant and compliant.
– *neuroticism* – reflects a tendency to experience anxiety and other negative emotions.

By setting up the different personality the agents in ALMA are able to feature different

affective reactions to the same events and actions.

Summary. ALMA provides us with three distinct types of affects – emotions for short term affect, mood for medium term affect and personality for long term affect. Personality is set up by user and affects emotions intensities as well as default agents mood. Mood is computed from emotions, which are decaying over the time (it takes around one minute for emotion to decay from maximum intensity to lowest value, but this depends on the setting).

4.3 ALMA GUI

ALMA features a GUI (Fig. 5) that can be used for a) observing the changing agents emotions and mood, b) setting up the agent's personality and c) setting the variables affecting emotions and mood computations (emotions decaying time etc.).
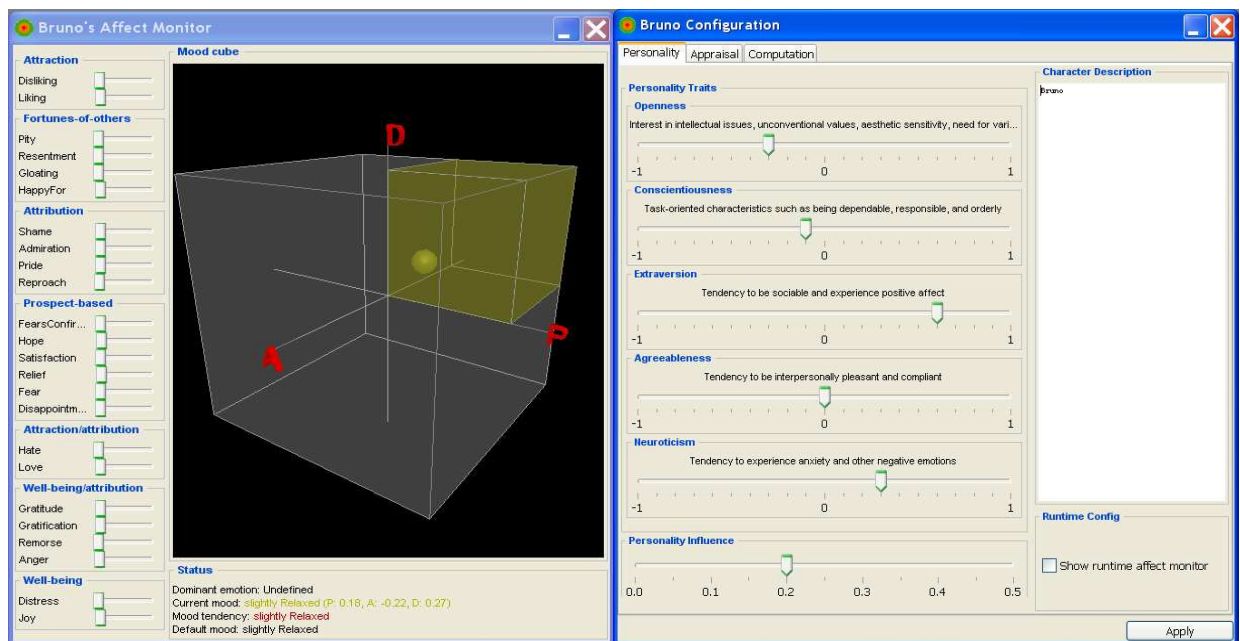
Fig 6. Two windows from ALMA GUI. Left window features ALMA monitor for observing the emotion and mood values. Right window shows personality configuration tab.

5 Project Emohawk overview

In this chapter we will present project Emohawk – a prototype implementation of Emohawk educational toolkit – in greater detail. If the reader is interested in Emohawk toolkit vision – in what do we want to have in the end – please read [Brom et al., 2009].

Project Emohawk is based on Pogamut 3 platform. This means that the agents decision making system is based on Pogamut sensory and act methods. As a virtual environment UT04 is used. For project Emohawk purposes we have slightly extended the Pogamut by new sensors and actors as well as added some content to UT04 to suit project Emohawk needs. Especially we have added non-violent (civil) models to UT04 game content to visualize our IVAs. To model agent's affect we have chosen emotion model ALMA, which was connected to Pogamut 3.

This chapter is organized as follows. First we will present current scenario content, then we will summarize interaction possibilities in the scenario. After this we will describe affect-driven architecture used for controlling the IVAs. Then we will speak a few words upon user-scenario interaction and lastly we will consider scenario story control features.

5.1 Scenario content

Project Emohawk scenario features:

- small town called UnrealVille – this is medium sized UT04 location (UT04 locations are also referred to as "maps") featuring four streets surrounded by buildings and square shaped park in the middle surrounded by a wall (Fig. 6). We would like to thank Gavin "Goslin23" Goslin and "Big Ed" for creating this UT04 map.

- one casually looking male IVA controlled by affect-driven architecture – Bruno (Fig. 3, left)

- two casually looking female IVAs controlled by affect-driven architecture – Anne and Clementine (Fig. 3, right). Note that both girls look the same now (currently we have found just one non-violent girl model).

- one alien looking scripted IVA – emohawk

For project Emohawk purposes we needed our environment to feature following places: a) cinema, b) park, c) Anne's home and d) Clementine's home. In UnrealVille there is no cinema and although there are many buildings, usually they are not accessible from outside. To solve this problem we have defined several points in UnrealVille that are perceived by our IVAs as places a), b), c) and d). The UnrealVille can be seen on Fig. 6 from bird's eye view with these places marked on.

Fig. 7. DM-UnrealVille bird's eye overview. On the picture we can see UT04 location DM-UnrealVille from birds eye. Numbers are marking locations of places we use in our scenario. 1. location of the cinema as well as Bruno and Anne starting place, 2. emohawk location, 3. Anne's home location, 4. Clementine's home location as well as her starting place, 5. meeting place in front of the cinema, 6. location of the park. Location DM-UnrealVille was created by Gavin "Goslin23" Goslin and "Big Ed". Copyright Epic Games 2004-2009.

5.1.1 Scenario affect expressions

Expressing the affect meaningfully is one of the problems the affective IVA development brings. In project Emohawk we used three ways of affect expression. Firstly, the affects values are effectively affecting agent decision making in the means of which action or proposal will the agent choose when interacting with some other agent. Secondly, the affects shape the casual conversation between agents (kinds of smilies are generated according the current agents affects). Moreover, thirdly around each agent head there is a set of flares floating around (seen on Fig. 3). These flares can change color, size, speed and a direction of movement. These flares are used to express agent current strongest emotions. To express emotion intensity we use the size (the higher intensity – the bigger flare). To express emotion type we use the color. We have mapped some of ALMA emotions to colors according to Table 2.

Emotion	Color
anger	bright red
fear	dark green
joy	yellow
distress	blue
love	pink
hate	black
liking	violet
disliking	brown
unspecified	white

Table 2. Mapping of emotions to colors. The emotions not found in this table will be visualized by white color.

5.2 Scenario interaction possibilities

IVAs in project Emohawk can interact with user and with each other using a) casual conversation, b) actions or c) proposals. We will now describe all the three interaction possibilities.

Casual conversation. When IVAs in our scenario meets another IVA or user they will start to speak with him. At the beginning they will greet the newcomer. After that the conversation is generated automatically. The strings contain smilies portraying IVA current state and attitude toward the other IVA. The delay between messages depends on whether the other is responding or not.

Actions. Time to time, but more often that not, the IVAs will select some action toward the one they are interacting with. The action selection is based on several aspects we will return to later in this chapter. In project Emohawk the actions are handled symbolically – there are no animations that would display them yet. Here we will present all the possible actions in the scenario:

– Compliment – Character A will say some positive things to character B.

– Kiss – Character A will kiss character B.

– Sex – Character A will make intimate action with character B and vice versa. This action is symmetric unlike the other actions (this means it is performed by both characters at the same time).

– Insult – Character A will say some negative things to character B.

– Slap – Character A will slap character B.

– Bye – Character A will end the interaction with character B in a neutral manner.

– Leave – Character A will end the interaction with character B in a negative manner.

– Cuddle – This action can be performed only to animals – emohawks. The character will cuddle the emohawk.

–   Kick – This action can be performed only to animals – emohawks. The character will kick the emohawk.

–   Proposal – Special type of action that is handled in a more complicated fashion. We will describe this in detail below.

Proposals. Proposals are a special type of actions. They allow the IVAs to negotiate certain things between them. First IVA makes certain proposal to second IVA, who may react to this either by accepting, rejecting or even ignoring the proposal (by not responding to this proposal at all). Each accepted proposal triggers certain behavior of IVA. Below we will outline all the proposals in the scenario as well as their triggered behavior:

–   Cinema – character A will propose character B to go to the cinema with him. If this proposal is accepted, the character A will lead the way to cinema followed by character B. Reaching the cinema location, the characters will start to watch the film – this lasts for 15 seconds, where the characters will not speak interact with anyone. After this the characters resume the interaction between them.

–   Park – character A will propose character B to go to the park with him. If this proposal is accepted, the character A will lead the way to park followed by character B. Nothing special happens when they reach the park place.

–   Home – character A will propose character B to go to home of character B. If this proposal is accepted, the character A will lead the way to character B home followed by character B. Nothing special happens when they reach the home place. However the home place allows for extended set of proposals to be selected.

–   Kiss – character A asks character B if he/she can kiss him/her. When accepted character A will perform action kiss toward character B.

–   Sex – character A asks character B if he/she wants to perform intimate action with him/her (possible just between opposite gender). When accepted intimate action occurs between character A and B. This is visualized by text messages in the environment.

–   Leave – Character A asks character B to leave all the other characters he/she may be interacting with right now. Accepting this proposal, character B will perform action bye to all the characters he/she is interacting with except character A.

We have outlined the possibilities of interactions in the project Emohawk here. Below we portray how the agent to agent interaction might look like in our scenario:

Example 1: Let's say Bruno wants to kiss Anne. To achieve this, Bruno will simply propose to Anne that he will kiss her. Anne receives this proposal and she will react to it for example by rejection. She does not want to be kissed right now. Bruno will be probably disappointed by Anne's reaction. Later on he may make the kiss proposal again perhaps with more luck.

Example 2: Let's say Bruno, Anne and Clementine have met at the cinema. Both girls like Bruno. So, lets say it will be Anne that will make a kiss proposal toward Bruno. Bruno accepts this proposal (he likes Anne as well as Clementine). Anne kisses Bruno with Clementine watching this. As a result Clementine will be angry toward both of them. She decides first to slap Bruno and then Anne. Bruno and Anne will not like this and they will probably respond with negative actions as slap, insult or leave.

We would like to note that currently all the actions our agents are performing –

kissing, slapping, watching film or making love are just schematically sketched using mainly text outputs. This indeed decreases the overall believability, but this bothers us not due to the fact that this is just a prototype implementation of our scenario.

Author's note. We would like to say that project Emohawk is just a prototype implementation of resulting Emohawk educational toolkit. Not all of project Emohawk actions and proposals are appropriate for educational purposes (e.g. sex action and proposal). These actions will not be included in the final Emohawk educational toolkit.

5.3   Agents control affect-driven architecture

Project Emohawk IVAs are controlled by finite state machine. They also feature affects provided by emotion model ALMA. The overview of Emohawk IVA architecture can be seen on Fig. 7. We obtain information about the environment through Pogamut sensors. These information are processed by agent state logic and may cause state transitions. The basic information from the environment is processed to create more complex percepts. These percepts are a) stored in agent's memory (if they contain some information about other agents) and b) processed by ALMA (if it is some kind of emotion triggering event).

The agent decision making is then affected by a) Pogamut sensory information, b) agents history (got from agent memory – history of interaction with the agent we are currently interacting with), c) ALMA affects and d) agent's current state.

Moreover our IVAs behavior is affected by rather large number of constants and timers – e.g. minimal delays between actions or proposals, intensity of affects needed to trigger certain actions, maximum waiting or maximum interaction timers etc.

Agent's memory contains information mostly about last actions and proposals, which took place between the agents.
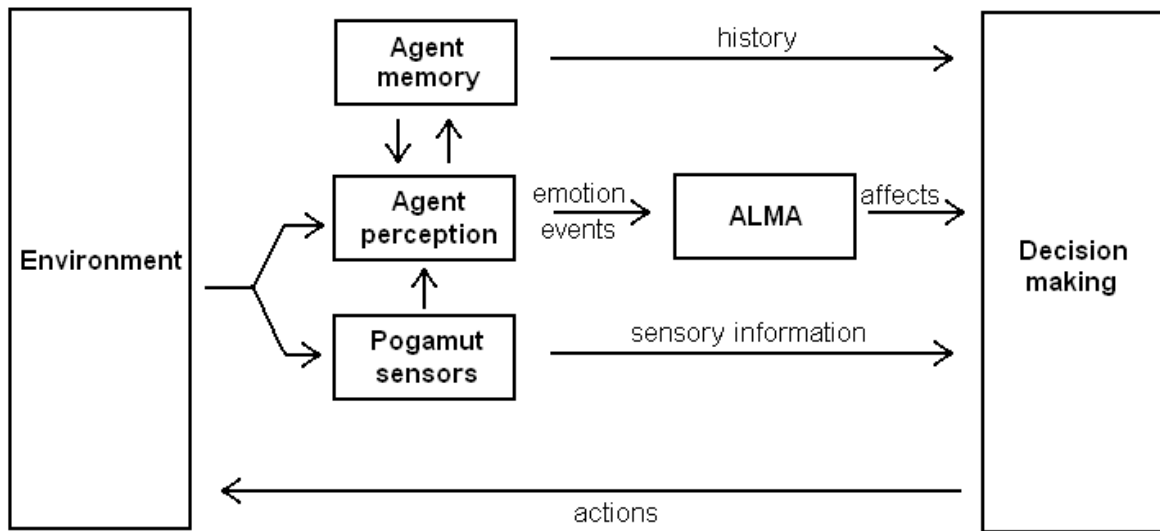
Fig. 8. Agent architecture. The information from environment are processed by a) Pogamut sensors providing our agent with basic sensory information and by b) agent perception module, that generates input for ALMA emotion model and stores relevant data into agent memory. The decision making is then affected by sensory information, agent history and agent affects got from ALMA.

Next we will present affect-driven architecture first by defining actual used agent affects and second by describing agent decision making system.

5.3.1 Agent affects

What we needed in project Emohawk was to somehow model agent attitudes toward other agents – the social affects. At the moment emotion model ALMA does not support social affects modeling explicitly. ALMA provides us with emotions, mood and personality. To model agents attitudes to each other we decided to use ALMA emotions, that can be naturally connected to agents or objects. Our effort resulted in a definition of new affect type – a feeling.

Feeling represents attitudes of agents toward other agents. These attitudes can be either positive or negative and differ in their magnitude. For feeling computation we picked two groups of ALMA emotions – positive group consisting of emotions love, like, joy and gratitude and negative group consisting of emotions hate, dislike, distress and anger. Positive emotions intensities raise the feeling value, while negative emotions decrease it. Some emotions have bigger weight than others in feeling computation (for example love weights more than liking). The emotions toward a character are obtained dynamically from ALMA engine – they change over time and so does the feeling value. Moreover emotions love, hate, liking and disliking are stored over the course and create solid part of feeling value. This solid part changes less rapidly over the time (normally the ALMA emotion decay from full intensity to zero in a minute or two).

weight	emotion
+ 0.5	gratitude
+ 1	joy
+ 1	liking
+ 1.5	love

weight	emotion
- 2	anger
- 1	distress
- 1	disliking
- 1.5	hate

Table 3. Feeling computation. This table shows the weights of emotions in feeling computation. On the left side positive emotions are shown, on the right side negative emotions are displayed.

The feeling computation equation according to Table 3 would be (the emotions are associated with the character we count the feeling intensity for):

Feeling int. = 0.5 * gratitude + joy + liking + 1,5 * love – 2 * anger – distress – disliking – 1,5 * hate

To summarize our agents uses ALMA emotions to compute emotional attitude toward other agents – a feeling. This feeling ranges from -5,5 to 4. Values above one can be considered as friendship, above two as love. Values below zero mean we dislike the target character. In project Emohawk we do not use any other ALMA affects at this moment although there are interesting possibilities for future work regarding ALMA mood and personality. Although the feeling values range from -5,5 to 4 the higher feeling attitude numbers will not last long (by higher we mean values above 2,5 and below -2,5). This is due to the reason that emotions in ALMA are decaying over time loosing the intensity. The maximum intensities of remembered emotions (liking, love, disliking and hate) can shift the

feeling value between -2,5 and 2,5.

We always store the maximum intensity of "solid" emotion our agent has experienced when interacting with other character. We exploit the fact that the "solid" emotions are recognized by certain emotional theories as complementary pairs. When our agent encounter new emotional maximum among one of the "solid" emotions, we overwrite this maximum and we also lower a bit the maximum of complementary emotion (emotion intensity – 0,3 * maximum of complementary emotion). The complementary pairs are love and hate and liking and disliking. This assures our model will be dynamic but at the same time the feeling trends will be preserved.

We illustrate how this works on an example. Let's say our agent maximums for emotion love and hate toward some other character are: love is 0,5, hate is 0,8. Suddenly our agent experiences emotion love to the character with intensity 0,6. According to our algorithm new love maximum will be 0,6 and new hate maximum will be 0,8 – 0,6 * 0,3 = 0,62.

The somewhat complicated algorithm for storing the solid emotions was developed because of the fact ALMA emotion model lacks listener to new emotions (although this listener is planned in future ALMA versions).

Below we will explain how the feeling is used in agents decision making.

5.3.2   Agent decision making

The project Emohawk agents are a finite state machines. The behavior they produce is dependent on:

a)   Agent's internal state (agent decision making state he is currently in, agent's history and agent's internal variables – constants and timers).

b)   Agent interaction status. What type of the interaction is our agent currently in.

c)   The feeling attitude towards the agent our agent is interacting with.

d)   Agent position in the environment. The agent will interact differently for example at cinema or at home.

Now we will say few words about on agent interaction status, followed by the description of agent states. After that we will describe in detail how our agents handle actions and proposals.

Agent interaction status. If our agent is interacting with some other agent, the type of this interaction is important. This type is determined by the status of the other agents. There are two possibilities. Either the other agent is treated as so called "agent with" or as an interrupter. The "agent with" will be always an agent of opposite sex. It is an agent we want to spend some time with and we will generally like him (the feeling value above zero). We can imagine "agent with" as someone we would go for a date with. If our agent is alone, he will consider the first agent he/she meets and that fulfills conditions above as the "agent with". The interrupter is an agent that will somehow interrupt our agent. If our agent already has an "agent with" and some other agent comes towards and speak to our agent, that agent will be considered as an interrupter. Also all agents of the same gender or with negative feeling value will be treated as interrupters.

Agent states. In our scenario we have two types of characters – boy and girls. For controlling these characters we use basically the same states with slightly different implementations for a boy and a girl. We will list the states of our agents here describing the behavior they produce:

–   **state agent alone** – this state handles a situation when the agent is alone in the environment. Implementation for a boy and a girl differs. The boy will start exploring the environment with a goal of meeting someone, while the girl just goes to her home. When exploring the agent randomly selects one point in the environment he/she will go to using path finding. Due to complexity of the location (town UnrealVille) we have picked around six points which can be selected by our agent and which are close to places we use in the scenario (cinema, park, home of Anne, home of Clementine etc.).

–   **state follow agent with** – this state is active when our agent is leaded by "agent with" somewhere. A typical situation – "agent with" proposed our agent to go to the cinema together. Our agent accepted and switched to this state. Now "agent with" is leading the way to the cinema with our agent following him.

–   **state going somewhere with** – this state is used when we are leading "agent with" somewhere. This time it was our agent who proposed to "agent with" to go somewhere. When "agent with" accepted our proposal, we have taken the initiative and switched to this state, which will make us to go to desired location with the "agent with" (we will wait for "agent with" to catch up).

–   **state interrupted** – this state is used when we are already with "agent with" or when we are doing some other activity in the environment and someone approaches us and speaks to us. We will switch to this state that handles the communication with multiple agents. Our agent switch his focus according to his feelings and according to last actions the other agents performed.

–   **state wait** – this state is active when we are waiting for someone. We will leave this state when the agent we are waiting for returns or when the waiting timer runs out.

–   **state with somebody** – this is the main state handling "agent with" interaction. In this state we pick proposals to the other agent. Which proposal will be picked depends on where we currently are (for example the sex proposal can be made just at home). Also we look on our proposal history. We will not propose to go to the cinema if we were recently there with the agent.

–   **state approach boys** – this state is used just by the girls. Since normally the girls just go home when they are alone, we have added this state that can be triggered after some time the girl is waiting at home. The girl will explore the environment when in this state searching for opposite gender characters approaching them and starting the interaction. There is a chance she will encounter Bruno with the other girl doing some things she would not approve raising the probability of conflict between the three agents.

The possible state to state transitions can be seen on Fig. 9.
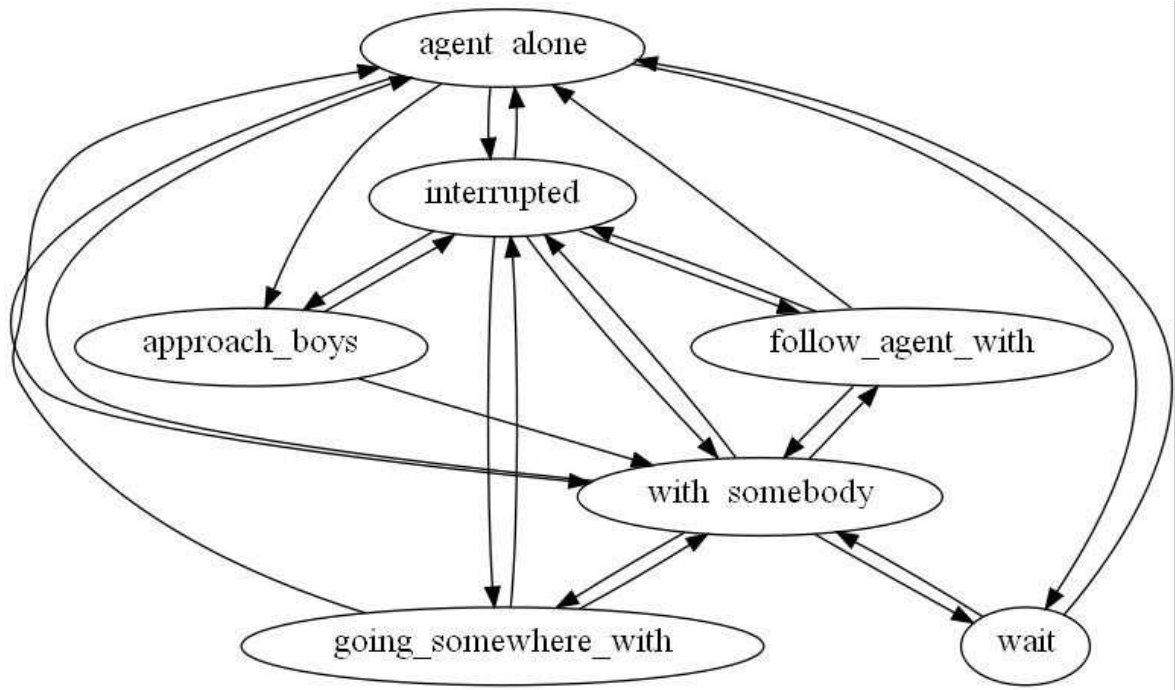
Fig. 9. Project Emohawk IVAs FSM transitions. On the figure we can see possible transitions between states for project Emohawk IVAs. State "approach boys" is reachable only for girls. Bruno and Anne begin the scenario in state "with somebody", Clementine in state "agent alone" (later she will go to near cinema location and switch to "wait" state waiting for Bruno). Emohawk is scripted and not controlled by states. Image created by GraphViz 2.0 [GraphViz, 2009].

Action and proposals handling. Here we will explain how and when our agents choose what action or proposal they will make, when interacting with some other agent. The most of the proposals in the project Emohawk scenario can be currently made just by Bruno. Only exception is proposal kiss that can be made also by the girls. Also we have forbidden Bruno to select action slap when interacting with girls.

There are two main states that enable our agent to make actions and proposals to other agents. These states are a) state "interrupted" and b) state "with somebody".

a) The state "with somebody" is the main state for "agent with" interaction. Here we will propose to "agent with" to go to the cinema, to go to the park etc. What we will propose and what action we will select depends on our current feeling towards the agent, location in the environment and our history (what we proposed or did to the agent in the past).

Let's say that Bruno is with Anne at Anne's home. Bruno could propose to go to the cinema, but since they were already at cinema, he won't do it. Now considering they are at Anne's home Bruno could make proposal to make love or to kiss Anne. However, Bruno's feeling toward Anne is not high enough for proposal to make love, so Bruno decides to make proposal to kiss Anne. If the proposal will be accepted the next Bruno action will be to kiss Anne. If rejected Bruno will repeat the proposal later (supposing his feeling attitude toward Anne will not change).

b) In state interrupted our agent generally will not make any proposals to anyone. But the agent is free to pick and do actions to "agent with" and to all the interrupting agents. Before the agent will actually do so, he needs to pick his focus. When picking the focus the agent considers the feeling value toward the agent and then if the other agent did some action or spoke to our agent recently. The agent will prefer other agents with higher feeling

32

associated with them and other agents that recently interacted with our agent.

Let's say that Bruno and Anne are on a way to Anne's home. They are going through park and suddenly they meet emohawk. The emohawk interrupts them by making adorable noises. Bruno and Anne turn toward emohawk and because they like him, they will probably both perform action cuddle. After that they will probably turn back facing themselves, because the feeling value toward each other should be higher than toward emohawk. However when the emohawk makes noise again or do something, they will probably turn back to him and interact with him again. After some time they will decide to continue way to Anne's home leaving emohawk alone (performing action bye towards him – maximum interruption timer will expire).

To summarize, action and proposal selection is dependent on:

a) Other agent status – "agent with" or interrupter? Our agents will not make proposals to interrupters.

b) Agent current state – agent will select actions and proposals for "agent with" in state "with somebody" and actions for "agent with" and interrupters in state "interrupted".

c) The agent's positions in the environment – agent will not propose to go to the cinema when at cinema. Agent will propose more intimate things when at home.

d) The agent's history – Agent will not propose to go to the cinema, when he/she was there with some other agent recently.

e) The feeling value – positive feeling value for positive actions such as compliment or kiss, negative feeling value for actions such as insult, slap or leave.

f) The constants – a set of constants defining needed feeling attitudes for certain actions and proposals to be selected as well as a set of constants.

5.4 User – scenario interaction

The user has two possibilities of using project Emohawk. He can either watch the story as a "ghost" spectator without solid body and without any influence on the story or he can join the scenario as a new character. As a ghost the user can observe the scenario floating around and watching the scene – the agents, the actions they are performing etc. As a new character, the user can communicate with any of the agent and perform any action toward him/her. The characters will react to him according to his gender (that is determined by users name) and according to the actions and messages he will use.

It could be possible for example to seduce one of the girls in the scenario considering the user is a boy or perhaps to seduce Bruno and make him angry on both the other girls considering the user is a girl. However the second goal would be probably much harder to achieve (it should be easy to seduce Bruno, while he is alone, but a bit harder to make him angry on both of the girls).

It would be even possible for two or three users to interact with our scenario at once as a new characters. This is the advantage of our robust system – our agents are reactive with just few things actually scripted.

We would like to note here, that we have not play tested our scenario with users interacting as new characters. Due to this fact some "evil" users would be probably able to achieve some non-realistic situation in the scenario to occur.

5.5   Scenario story control

The story controller in current project Emohawk implementation is coded directly into agent reactive behavior. This means at some points we check several conditions that may lead to state or behavior change. This rather "dirty" approach can be justified by the fact, that currently we model just one simple story. For Emohawk educational toolkit we will require story controller able to control the characters in a way, they will be able to produce several different stories. This story controller should be configurable and able to control the scenario story by changing characters goals and internal variables. Currently we simulate this controller by our "dirty" reactive approach.

We will start this sub section by describing a usual scenario story, then we will consider the story control and answer the question where does the emergence in our scenario come from.

A usual story. Here we will describe a usual story that is produced by project Emohawk in a greater detail.

The scenario starts with Bruno and Anne at cinema with both of them remembering they were there together just a minute ago. Their feeling attitude toward each other is preset to a higher value (around one). The result of this and of the fact they are close the each will be, they will select each other as "agent with". The boy in our scenario is the one who is making the most of the proposals. Due to the higher feeling values Bruno will a) perform positive actions as compliment, b) make a proposal to Anne to go to Anne's home. How quick this will happen depends a bit also on chance. Anne will usually accept the proposal to go home with Bruno due to higher feeling value (this value is probably further increased by any positive action Bruno performed). The result of this first encounter should be Bruno and Anne going to Anne's home.

Meanwhile Clementine is waiting at her home, she will eventually start to walk towards cinema, where she should meet Bruno. If the Bruno and Anne take long time to decide what to do, she may be even able to catch them there. Otherwise, she will just wait at the place close to cinema for Bruno to appear. If he will not make it in time, the Clementine will be disappointed (feeling toward Bruno decreases) and will go home again.

Let's return back to Bruno with Anne on their way home. In the park they will be walking around emohawk. Emohawk may or may not interrupt them – this depends on whether he makes a noise, when Bruno and Anne are passing by (this is more or less random). If the emohawk interrupters them, Bruno might not make it in time to meet Clementine.

After the emohawk encounter, Bruno and Anne end at Anne's home. They feeling value should now be considerably higher than at the beginning of the scenario as the result of all the positive actions and accepted proposals. Bruno will now probably propose either kiss or even to make love. If the kiss goes first, it considerably increases the feeling value (if accepted). The next Bruno's proposal will be with high probability to make love. After the intimate action, which is represented by text messages, Bruno will say goodbye to Anne and will go to cinema to pick up Clementine. It is worth noting that Bruno will say goodbye and walk to the cinema, even if Anne rejects the proposal to make love. This was scripted to help Bruno to make it in time to cinema.

If Bruno arrives at cinema and Clementine is already gone, Bruno will start exploring the environment. He will switch to "agent alone" state. In this state he will approach any girl he sees. He may even find Clementine – if it happens he will walk toward her and start the interaction. Even though Clementine was a bit disappointed by Bruno not arriving in time,

Bruno still may be able to win Clementine back by positive actions and conversation (this also usually happens). They can end up at cinema and eventually at Clementine's home. However if Bruno makes it on time, he will suffer less problems persuading Clementine to do stuff with him.

Meanwhile Anne will stop waiting at her home (her waiting at home timer expires). She starts to explore the environment. She may meet Bruno with Clementine at cinema or somewhere else. This will probably result in a conflict, because the girls are generally jealous if they see Bruno doing some positive actions towards some other girl. We would like to note here that our agents are currently naive when it comes to action selection in front of other agents (Bruno will naively make positive actions toward a girl even if the other girl is near resulting in him being caught). The conflicts obtained by this, may result in negative feeling shifts. Negative feeling values promote the selection of negative actions that may now occur between characters. All of this may result in one or both of the girls leaving.

When Bruno is left alone, he starts to explore environment again and will interact with any girl he still has a positive feeling attitude to. However if a situation would occur in the scenario that all of the characters would have negative feeling attitudes toward each other, no further interaction would occur between them. This almost never happens in our scenario.

The story continues featuring other agent encounters and finishing after 10 minutes from start. This story was created solely for testing and evaluating purposes, story for Emohawk educational toolkit will differ in many aspects (e.g. length of the story, narrative curve, list of possible actions and proposals etc.).

Story control. Bruno's ordering of proposals supports simple story generation. Bruno first tries to make a cinema proposal, after going to cinema and watching film, he proposes to go home. At home he may propose to kiss or to make love. When Bruno was recently with a girl at cinema or at home, he will not make these proposals to her. All of this can result in sequences as:

Bruno meets girl somewhere, he takes her to cinema. After that he will go home with her, where he proposes more intimate things to her.

In our scenario, girls cannot make proposals except the proposal to kiss someone. Now we will answer the question:

Where does the emergence come from?

The emergence. The emergence in our scenario is hidden in agent to agent interaction. The story emergence is supported by random components of our affect-driven architecture. We generate them ourselves (e.g. by adding some small random values to certain agent timers) or the platforms we use generate them for us (Pogamut uses TCP/IP protocol to communicate with UT04 – that may cause small unpredictable delays in the communication). Another random component is the behavior for exploring the environment. Our agents pick randomly their destination – so when Anne starts exploring the environment from her home, we cannot know how fast she will find Bruno and Clementine. She even may miss those two completely resulting in a peaceful conflict-less scenario.

These random components have an influence onto our agent's affects – that is the feeling value – the key component for our agent's decision making. For example lets say Bruno will experience two stacked positive emotion events from Anne. This may raise Bruno's feeling value to Anne to allow proposal kiss to be selected. Successful action kiss

between those two will have further positive effect on the feeling, completely changing the shape of the interaction. This can further influence the story by Bruno and Anne leaving the cinema sooner.

6   Implementation

In this chapter we will dig into more technical details of project Emohawk implementation. The implementation as well as this chapter is divided into two main parts – 1) implementation in UT04 and 2) implementation in Pogamut 3.

6.1   UT04 part

UT04 is an action game that does not natively support non-violent scenario. Since we wanted to use UT04 as a base for project Emohawk, that is non-violent, we needed to extend the content of UT04 to allow this. We have started this work by searching the large UT04 on-line cache of game extensions and modifications until we have found – UT04 location called DM-UnrealVille and two civil models of IVA for UT04 – a man and a woman. We have added this new content to the game and modified GameBots to allow the use of civil models.

Moreover we have added some new interaction possibilities for IVAs in GameBots. The IVAs are now able to give items to each other and they can perceive this as well.

Concerning the emotions we have implemented a custom emitter in UT04 that allows for visualizing and controlling of a set of flares that can float in the environment. These flares can change dynamically direction of movement, size and color. Currently we use them for expressing the agent's emotions (Fig. 3).

6.2   Pogamut 3 part

Concerning the implementation in Pogamut 3, we needed to create agent decision making system, program the scenario story, connect emotion model ALMA and implement the methodology for scenario analysis. Here we will speak firstly about the Pogamut decision making and the way our affect-driven architecture was implemented. We will also concern the story management that is hidden beneath it. Secondly we will describe how are the interaction between agents implemented. Thirdly we will concern the mapping of environment events to emotion model ALMA and lastly we will speak about scenario analysis implementation.

6.2.1   The decision making

The Pogamut agent's decision making is based on a *doLogic* method. This method is invoked each time a synchronous batch comes from GameBots containing information about agent's surroundings. This happens every 250 ms. Through this method we are emulating simple finite state machine system. Each time this method is called, we look in which state we are currently in and execute the state code hidden in another method. This results in our agents are able to produce some actions, proposal or conversation each 250 ms. We have also implemented *beginState* and *endState* methods as well as *gotoState* method – used for state transitions.

Our agents have implemented methods for action and proposal selection. These methods concern agent internal state – this includes a set of timers. Our agents store information about the time they performed each action to someone. This is then used by the methods not to pick certain action or proposal too often. It may happen that this methods return null – this means the decision is to do nothing to target agent at this time.

Each proposal has some feeling value that has to be exceed in order for character to

select it. When more proposals are free to be chosen, there is a certain order our characters will tend to use. Bruno will first select the cinema proposal, then park proposal, then home proposal. At home he can make kiss and sex proposals.

Also when Bruno is at cinema, he will try to make home proposal as his next. This simple ordering produces a simple story sequences where Bruno is taking girls first to cinema and then home, where he proposes more intimate things (however agents are permitted to kiss also elsewhere). Set of timers prevents Bruno from selecting a proposal, when he did the proposal thing with some other agent recently. This approach allows Bruno to react at any time to any other agent with proper selection of actions towards him.

6.2.2   Agent interactions

All the communication between agents and between user and agents is done through UT04 text message channel. This includes also action and proposal interaction. A simple interaction between two agents, where Bruno proposes a kiss between him and Anne, who accepts, looks like this:

Bruno: "To:Anne, making proposal kiss".

Anne: "To:Bruno, proposal kiss accepted".

Bruno: "To:Anne, ACTION KISS"

The communication between agents is handled through a set of keywords. The "To:" keyword followed by a set of names separated by commas determines the agents this message is for. If the "To:" keyword is missing the message is considered to be for everyone. If there is "To:" and no names are after, the message will not be parsed by anyone. Note that the agents need to see the one who is speaking to them in order to receive the message.

Now to determine if the message is an action, a proposal or simple text message another set of keywords is used. This set includes:

– "proposal" – followed by one of the proposal types. Marks that this message is a proposal of input proposal type. To further determine if the agent is responding to some proposal or is making a new proposal, one of following three keywords is added: "making", "accepted" or "rejected"

– "ACTION" – followed by one of action types. Marks that this message is an action of input action type.

If the message does not include any action or proposal keyword, it is treated as a casual conversation. For casual conversation agents parse a set of smilies to evaluate the conversation emotional value. Following smilies are parsed in the text messages:

– ":-)" and ":-(" – substitutes emotions joy and distress. More positive ":-)" smilies mean the emotion joy will be generated upon receiving this message. The intensity depends on the number of smilies. The highest joy intensity will occur if the result is that in the message there is five more positive smilies than negative ones.

– ":-*" and ">:@" – substitutes emotions liking and disliking. More positive ":-*" smilies mean the emotion liking will be generated upon receiving this message. The intensity depends on the number of smilies. The highest liking intensity will occur if the result is that in the message there is five more positive smilies than negative ones.

– "#!" and ":-O" – substitutes emotions anger and fear. More "#!" smilies mean the emotion

anger will be generated upon receiving this message. The intensity depends on the number of smilies. The highest anger intensity will occur if the result is that in the message there is five more "#!" smilies than ":-O" smilies.

Our agents are generating the smilies in the text messages according the actual emotions joy and distress toward the target agent (smilies ":-)" and ":-("), according the actual feeling value (smilies ":-*" and ">:@") and according the emotions anger and fear (smilies "#!" and ":-O").

6.2.3 Agent perception (emotion events)

Here we will consider the mechanism of agent perception of the environment as well as technique we use for mapping environment events to ALMA inputs.

The agent perception is based on Pogamut sensory methods and Pogamut listeners. Sensory methods allow are agent to ask for example: "Do I see any player right now? And if yes, who is it?" The listeners allow our agents to listen to certain message types that bring information about certain environmental events. For example we can "listen" to all of the text messages our agent receives.

These methods and listeners provide us with basic information about environment – what do we see or what we recently heard. However, since we wanted our agents to be able to react to various types of situations emotionally, we needed to built upon these basic methods to provide our agent with more complex percepts. For example we are not interested just if we see someone, but we would like to know how long we are seeing him, when was the last time we saw him, if we actually like him and so on. We solved this by post-processing some of the environment events. This provided our agents with more complex percepts. Using these our agents are able among other things to recognize pre-defined set of events affecting their emotion state. These events are appraised by a set of ALMA emotion variables and processed by ALMA. This can result in a change of agent's emotions and mood. Our agents are able to recognize following set of emotion triggering events:

– *action kiss event* – our agent kissed someone
– *action kiss other event* – some other agent kissed someone
– *action kiss by event* – someone has kissed our agent
– *action slap by event* – someone has slapped our agent
– *action slap other event* – some other agent slapped someone
– *action kick other event* – some other agent has kicked someone (should be animal)
– *action kick by event* – someone kicked us – should happen just to animals
– *action bye by event* – someone said bye to our agent
– *action bye other event* – some other agent said bye to someone
– *action compliment by event* – someone said compliment to our agent
– *action compliment other event* – some other agent said compliment to someone
– *action cuddle by event* – some other agent cuddled us – should happen just to animals
– *action insult by event* – some other agent insulted our agent
– *action insult other event* – some other agent insulted someone
– *action sex by event* – some other agent made love with us
– *action sex other event* – some other agent made love with someone
– *action leave by event* – some other agent left us (said goodbye in a bad way)
– *action leave other event* – some other agent left someone (said goodbye in a bad way)
– *alone event* – generated once in a while when our agent is alone
– *pickup event* – our agent has picked up some scenario item

- *player lost event* – our agent has lost sight of some other agent
- *player together event* – generated once in a while when our agent is together with someone else
- *player appeared event* – our agent sees some other agent, who was not seen in some time
- *anger fear message event* – our agent received a message containing anger and fear smilies
- *happy sad message event* – our agent received a message containing happy and sad smilies
- *like dislike message event* – our agent received a message containing like and dislike smilies
- *message to other event* – some other agent send message to someone
- *conversation ignore by event* – some other agent is not responding to us
- *interrupted event* – we were doing something and were interrupted by someone
- *received item event* – we have received an item from other agent
- *item jealousy event* – some other agent received item
- *proposal to other event* – some other agent made proposal to someone
- *proposal to other by agent with event* – "agent with" made proposal to someone
- *proposal other response event* – some other agent responded to proposal by someone
- *proposal response to other by agent with event* – "agent with" responded to proposal made by someone
- *proposal by other to agent with event* – some other agent made proposal to "agent with"
- *proposal response event* – some other agent responded to proposal we have made to him
- *proposal ignore event* – some other agent ignored proposal we have made to him
- *received proposal event* – some other agent made proposal to our agent
- *polymorph action bite event* – emohawk bit our agent
- *polymorph dislike fear event* – emohawk is making bad noises
- *polymorph like event* – emohawk is making lovely noises
- *wait agent not returned event* – our agent was waiting for someone and he did not return
- *wait agent returned event* – our agent was waiting for someone and he returned
- *wait in progress event* – generated once in a while, when our agent is waiting for someone

We appraise these events by ALMA EEC variables desirability, praiseworthiness and appealingness. The intensities may depend on actual feeling attitudes to agents that were the causes or targets of emotion events. More information can be found in project Emohawk programmer's documentation.

6.2.4 Scenario analysis

Each scenario run (experiment) lasts for 10 minutes. When the experiment ends we store its progression in Java binary files. We have three files for each experiment – one for each main character of our scenario – Bruno, Anne and Clementine. In these files all the events, actions, feeling shifts and important aspects of agent's internal state are stored. Later by processing these binary files we create a text and graph output portraying the aspects of the experiment we are interested in.

For plotting of all experiment graphs we used free Java library jFreeChart [JFreeChart, 2005-2009]. To evaluate the text files we used regexp matching.

Sub-scenes. For the analysis purpose and better orientation in scenario story, we have defined a concept of sub-scenes. Each scenario run (we will refer to scenario run as to experiment

later on) is automatically divided for each character into a set of sub-scenes according to characters internal state. These sub-scenes create a sequence that can be later used to "tell" the character's scenario story. For example one story sequence in our internal representation for Bruno one story sequence may look like:

WITH_ANNE_AT_ELSEWHERE;WITH_ANNE_ON_WAY_TO_GIRL1_HOME;WITH_ANNE_AT_ANNE_HOME;AGENT_ALONE_AT_ANNE_HOME;AGENT_ALONE_AT_ELSEWHERE;WITH_CLEMENTINE_AT_ELSEWHERE;WITH_CLEMENTINE_ON_WAY_TO_GIRL2_HOME;WITH_CLEMENTINE_AT_CLEMENTINE_HOME;AGENT_ALONE_AT_CLEMENTINE_HOME;AGENT_ALONE_AT_ELSEWHERE;WITH_ANNE_AT_ELSEWHERE;WITH_ANNE_ON_WAY_TO_GIRL1_HOME;AGENT_ALONE_AT_ELSEWHERE;WITH_CLEMENTINE_AT_ELSEWHERE;WITH_CLEMENTINE_ON_WAY_TO_CINEMA;WITH_CLEMENTINE_AT_CINEMA;WITH_CLEMENTINE_ON_WAY_TO_GIRL2_HOME;WITH_CLEMENTINE_AT_CLEMENTINE_HOME;AGENT_ALONE_AT_CLEMENTINE_HOME;AGENT_ALONE_AT_ELSEWHERE;

We could read this internal story sequence for example as follows:

"I was with Anne, we have chatted a bit and then I walked her home. We were at Anne's home for a while. Then I departed leaving Anne at home. After some time I have met Clementine somewhere. I walked Clementine home as well. I have spent some time there and then departed. Then I met Anne again. We ended up at Anne's home again. After some time I met Clementine. Together we went to cinema. From cinema I walked Clementine home. We spend some time there and then I said goodbye to her and went out for a walk."

Note that some aspects are repeating in our story. Currently this is not a problem because of our project being prototype implementation. We have purposefully set the scenario to run for 10 minutes to see more of possible interactions between agents. For Emohawk toolkit the scenario story will be modified.

The sub-scene generation algorithm is based onto agent's internal state, agent's current position in the environment, name of agent's current "agent with" and agent's goal location if any. The sub-scenes identifiers will be described in more detail in following chapter that will also present the project Emohawk evaluation.

Note. For those interested in technical details we present here a brief summary of current scenario evaluation demands. To analyze our scenario we have produced more than 12 000 text and image files. The data of all the 357 experiments did not fit in 4 GB of memory. The generation of output files lasted more than two hours on Intel Core2 Duo 2.4GHz processor.

7 Evaluation

An important part of the work on project Emohawk was to analyze its performance in the means of believability and the story unfolding. The story outcome the project Emohawk produces is partly emergent. This makes the analysis more complex. We have run the project Emohawk scenario multiple times (357 experiments). This was necessary to see the most of possible story branches in the scenario. Since it was not possible to watch every run of the scenario (one run lasts 10 minutes), we have logged the scenario progression and performed the analysis afterward.

The problem with emergent narrative is that not all of the outcomes of possible interactions between IVAs can be fully anticipated due to the fact that the story is not scripted. There may occur interactions conflicting with believability or the whole story may unfold in most cases differently than what we would like to have. Due to these reasons it is more important to commit a thorough analysis of such system. In fact analysis of these emergent systems is at least as important as the implementation itself. Without the analysis the developer does not really know, what the system actually produces.

We would like to note two things here. Firstly, the evaluation of emergent narrative scenario will never be 100% accurate. Especially there are problems connected with defining "a good story" or "good" agent to agent interaction. Secondly, as project Emohawk is just a prototype our evaluation presented here is also a prototype evaluation as well. Some aspects of our scenario were designed in a way that made them not exactly believable (e.g. repetition of actions, proposals and sub-scenes sequences etc.). This bothers us not at this moment. The list of issues we will need to solve for Emohawk toolkit to be ready will be presented in Chapter 8. This evaluation served the purpose of validation. It is able to capture aspects of the scenario we treat as suspicious in our prototype version right now.

In this chapter we will explain how we have evaluated scenario runs (we will refer to them as to experiments). First we will introduce the used methodology, then our tools, next we will present our results, afterwards we will analyze suspicious experiments in a greater detail and lastly we will provide a summary of all the results here.

7.1 Methodology

For the project Emohawk purposes we have created a basic methodology that allows for emergent experiments analysis. This analysis should:

a) analyze the emergent potential of our scenario

b) analyze the believability of the story outcomes and of the agent's behavior

The creation of this methodology is an ongoing process. Our methodology indeed does not capture all the possible problems that may occur in the scenario, but we will show it can capture at least some of them. Our methodology produces a summary results from high number of experiment runs, that can be reviewed by the developer to pinpoint potential problems. We have considered just unintended features of our prototype scenario.

To evaluate project Emohawk performance we have used following criteria:

1) reliability of communication between IVAs

2) the feeling value development

3) actions and sub-scenes occurrence

4) actions and sub-scenes sequencing

5) sub-scenes durations

Now we will explain these criteria one by one.

7.1.1   Reliability of communication between IVAs

When analyzing the experiments we have noticed a rather undesirable event to occur. Some of the text messages between IVAs were not received by the target agent (10% – 20%). After investigating this issue we have found out it is a feature of the new Pogamut 3 core that is currently in beta stage of its development. We have decided to include this phenomen in our analysis, because surprisingly, this issue had also some positive effects on the scenario performance. We will analyze this by plotting the statistics of how many of proposals, actions and conversations between agents were actually lost.

7.1.2   The feeling development

Project Emohawk IVAs feature affect-driven architecture. This architecture is a key component in respect of emergent behavior produced by our project. Hence one of our goals was to evaluate this architecture in the means of believability and emergent behavior. As the feeling value toward other agents is critical in determining what interaction our agents select it received more than moderate attention. Below we present what aspects of feeling development were in our analysis scope.

1) "Was it a happy ending?" – considering the feeling relations at the end of the experiment

We needed some measures to evaluate how the experiment ended. One of our measures is the end feeling value for all the three characters towards each other. We asked questions as: Were Bruno and Anne lovers at the end of the experiment? Did Clementine hate Anne? And etc. We have defined that the end feeling value above two meant "lovers", above one "friends", above zero "positive" and below minus one "hate". We were interested if the outcome in the means of end feeling values will be mostly the same for all the experiments or not to determine project Emohawk emergent story potential.

Our other concerns were: Let's say Bruno loves Anne, does she love him back? Is it believable when Bruno has high end feeling towards Anne, but Anne hates him? How often does this happen? We needed to inspect these situations where feeling relations asymmetries occurred. These are surely less standard situation, we should analyze in greater detail.

We created a text output for each experiment that contained the information about the end feeling values for all of the agents. This allowed for easy analysis of this aspect.

2) "You are angry? Well I am as well!"– considering the feeling development of characters toward each other

We were also interested in feeling development during the whole scenario – especially in "conflict" occurrences. Our simple definition of these conflicts was a rapid decrease in the feeling value for one or more agents toward other agents. We were interested in asymmetries in feeling development during the experiment.

3) "Hit me! I will smile." – considering the feeling development after emotion events in the experiment

43

The believability of our characters and emotion reactions was one of our concerns as well. To see the performance of our architecture we analyzed the feeling values development for a character one second after the character experienced an emotional event. We would expect that the negative emotion events will tend to lower the feeling values at least to some of the other characters, while the positive events would raise it. This is not that simple because multiple events could have both positive and negative feeling impact, but we were more interested in the general trend. We measured the emotion events feeling impact by the difference between feeling values at the time of event and one second after the event.

4) "I am always happy when you are at home..." – considering the feeling development in the sub-scenes

Considering the sub-scenes in our scenario we asked a question – are there some sub-scenes with higher conflict probability? Are there other sub-scenes during which the characters feelings will tend to raise? Our answer to this question is yes. The hypothesis is simple: Generally the sub-scenes featuring all three characters will produce more conflicts, hence more feeling decreases than in sub-scenes featuring just two characters. The sub-scenes featuring a boy and a girl will tend to have feeling increments.

To analyze sub-scenes feeling development we focused on two things. First on the feeling difference – difference between feeling value at the beginning of the sub-scene and the end of the sub-scene. Second we were interesting in the feeling value for the characters at the end of the sub-scene. We used box plot graphs to analyze this.

7.1.3   Actions and sub-scenes occurrence

Above we used end feeling values to determine how the experiment has ended. To get a better grasp of what has happened during the experiment we focused on whether certain actions and/or sub-scenes took place in it. We have defined several key points in the terms of actions and sub-scenes and we watched how often they occur in the experiments. This further helps us in the determining our project Emohawk emergent story performance. Moreover we were interested in global actions' statistics – e.g. how many times and toward whom Bruno performed action kiss in all of the experiments. Considering the sub-scenes – we were interested if certain sub-scenes tend to contain certain actions. For example sub-scene "agent alone" containing action kiss would be highly suspicious.

Note that there is a difference between evaluation and testing of the project. The evaluation is more concerned with emergent outcomes of the scenario, while testing tries to pinpoint potential problems in the implementation. Sometimes it can be hard to separate these two things. The question is: "Is this an emergent feature or scenario implementation problem?". Our evaluation was able to capture also some scenario implementation problems.

We have defined two key points for our scenario analysis purpose.

1) "Clementine, have you been at cinema today?" – considering the sub-scenes occurrences

Considering the sub-scenes, we were interested especially: Was Bruno at home with both of the girls, just one or none? And: Was Bruno at cinema with Clementine? We will not ask if Bruno was at cinema with Anne – he always was, because the story starts with him and Anne at cinema. To analyze this we process output text files.

2) Intimate action in the scenario.

When the feeling value is high enough, our agents can perform intimate action (action sex),

when they are at home. This intimate action has the highest requirements on the feeling value between characters, so we have decided to include brief analysis concerned with its occurrence.

7.1.4 Actions and sub-scenes sequencing

"I'll kiss you after I slap you!"– considering the actions and sub-scenes pairs

Considering again the believability of our model we have conducted an analysis focused on the sequence of actions and sub-scenes in an experiment. We will call a pair a sequence of two actions or sub-scenes that come after. Certainly, there are pairs of actions and sub-scenes that does not seem plausible or believable. We analyzed our scenario in respect of these so called suspicious pairs of sub-scenes and actions.

Firstly, we got a list of all the pairs of actions and sub-scenes, which occurred in the experiments. Secondly, we processed this list and defined a list of suspicious actions and sub-scenes and analyzed in detail the experiments that contained these pairs. This enabled us to find out whether our model failed, or if it is just a rare feature of experiments that does not break believability.

"First I went home with Anne, then I returned to cinema for Clementine, after that I..."– considering the sub-scenes sequencing

We were also interested how many sub-scene sequences will project Emohawk produce. Will there be just a few of them? Or will they differ from each other in each experiment? To analyze this we simply count the occurrence of each found sub-scene sequence for all the characters in the experiments.

7.1.5 Sub-scene durations

"Then I was at home with her for five seconds."

Lastly we have plotted the durations of sub-scenes to see if there are some major abnormalities. We would generally expect smaller differences between the sub-scenes durations, although we do not have any hypothesis to test in respect to this. Also we searched blank places – times when the characters did not have any sub-scene defined. If this would occur too often, our algorithm for sub-scene division would be faulty.

7.2 Tools

Here we will present what tools we have used to perform the analysis described above. Methodologically we have used two types of tools. Firstly, automatic algorithms that processed the experiments and created an output – text files and graphs. Secondly, we used our human judgment to process the outputs of the automatic analysis to determine a) suspicious experiments intended for detailed analysis and b) problems of experiments connected with believability and emergence.

Each experiment produced more than 16 text files and 19 graphs (some of those were multi-graphs). Moreover we got more than 25 other graphs and text files for analysis of all experiments as a whole. Below we will provide a brief review of outputs we used.

7.2.1   Used graphs

Here we will present a selection of the graphs we used for project Emohawk analyze. For each character in an experiment we used several graph types to observe what has happened to him during the experiment. Moreover, each experiment produces one graph used to overview feeling development for all the three characters at once. To analyze all the experiments as a whole, we produced box plot graphs showing overall statistics. Below we will some of the graphs we used.
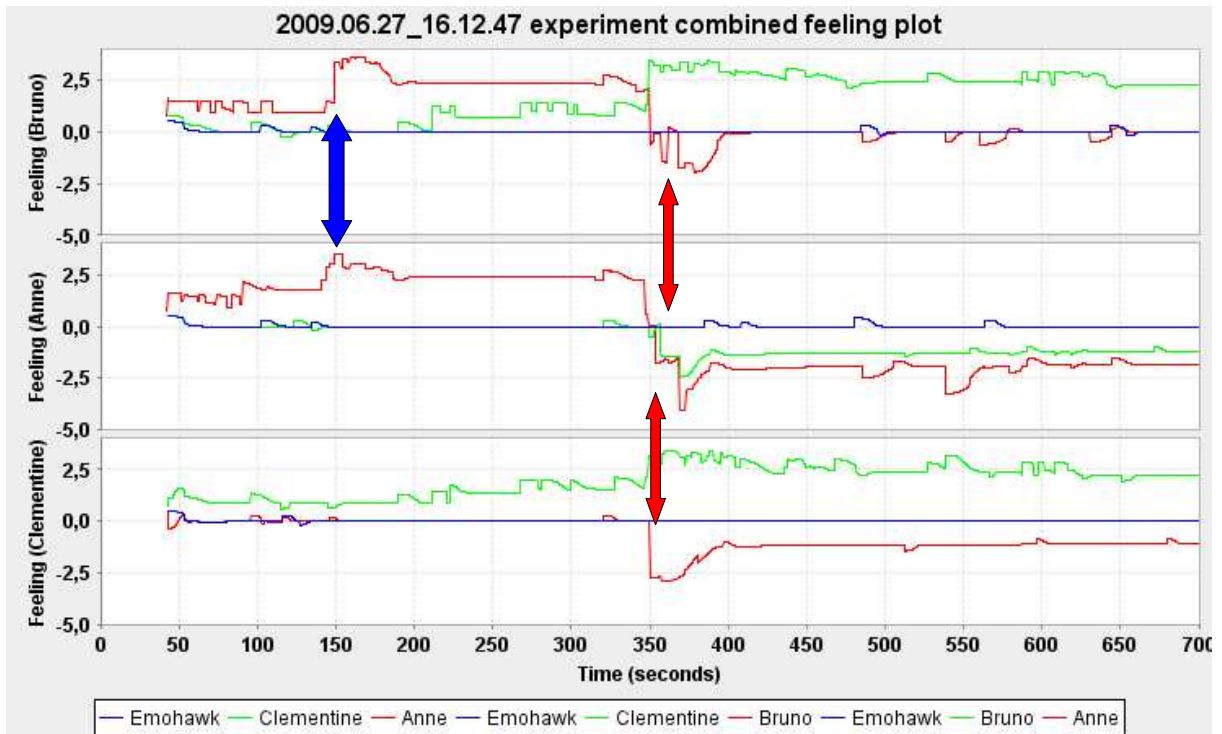
Graph 1. Combined feeling plot. This graph shows the feeling development for all the three characters in a usual experiment (2009.06.27_16.12.47). Upper graph plots feeling development for Bruno, middle graph for Anne and lower graph for Clementine. Emohawk is represented by blue line in all three graphs. Bruno is represented by red line in the middle graph and by green line in the lower graph. Anne is represented by red lines in upper and lower graph. Clementine is represented by green line in the upper and middle graph.

To plot the feeling development of all three characters during the experiment we used Graph 1. On Graph 1 we can observe a feeling increase between Bruno and Anne (marked by blue arrow). This can be the result of some positive actions performed at Anne's home (guessing from the time of the increase). We can also see a conflict on the graph marked by a red arrow. This conflict occurred between Anne and Bruno and Anne and Clementine. Regarding the emohawk we can see that nobody had interacted with him in this experiment – the feeling attitude towards him remains most of the time at zero.

46
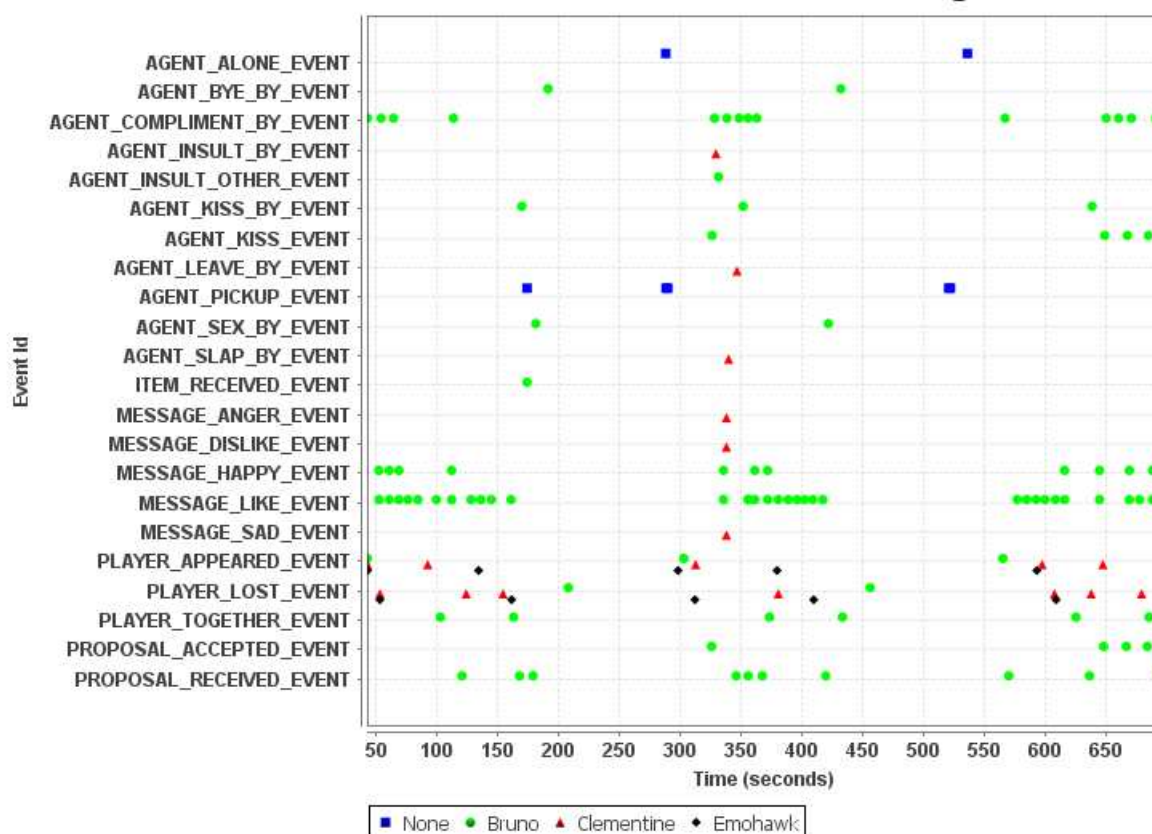
Graph 2. Emotion events perceived by one of the characters. The graph is showing emotion events that were perceived by Anne in experiment 2009.07.13_22.43.17. The causes of these events are represented by color and geometrical shapes. Green circles are events caused by Bruno. Red triangles are events caused by Clementine. Black quadrangles are events caused by Emohawk and blue squares are events that were caused by none of other agents.

Graph 2 is showing emotion events experienced by one during the experiment. This graph can be used to find out what event caused the feeling shifts in feeling development graph.
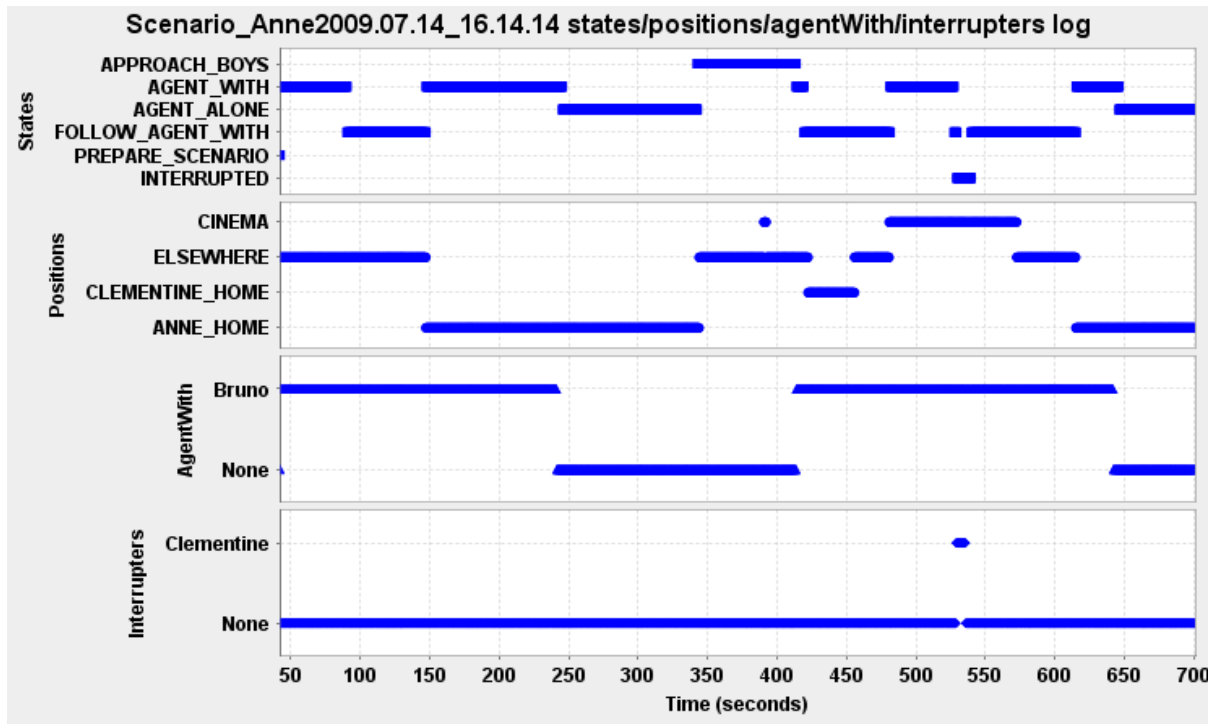
Graph 3. Internal state of one of the characters. This graph shows internal state of Anne during experiment 2009.07.14_16.14.14. The upper graph is showing the states of agents logic, the upper middle graph is showing the location in the environment the agent was at, the lower middle graph shows "agent with" name and the last lower graph shows the interrupters names.

We were also interested in our characters internal state. What state they were in when interacting with the other characters? Were they at cinema or at home? Did someone interrupt them? Graph 3 is plotting agent's internal state, agent's position in the environment, agent's "agent with" name and all the interrupters names. The algorithm for sub-scene generating is based on these internal variables.

Considering the analysis of all of the experiments we used box plot graphs showing:

– the feeling development after certain emotion event occurred

– the feeling differences during sub-scenes

– the feeling values at the end of sub-scenes

– the sub-scenes durations

We will describe these graphs in the result section of this chapter.

7.2.2 Used text files

Following text files were used to further evaluate the experiments:

– sub-scenes for one character with their duration and feeling differences

– files with all the proposals and actions with their times and feeling value toward the character they were done to

– files with information about all the proposals the agent made and their results – accepted, rejected or ignored

48

- files containing the feeling development for the character

- files containing all the emotional events with their times and values of OCC variables

- three files containing the results of action pair analyze

- three files containing the results of sub-scene series and pairs analyze

- files containing the end feelings of all the characters towards all the other characters

- files containing the count of values that were used to plot emotion events feeling difference graphs and sub-scene feeling difference and end values graphs

- files containing the information about action counts during sub-scenes, whole experiment

- files containing the information about missed action counts – caused by the message loss problem

To process these files we used often Regexp matching. Tables in the following sub section of this chapter will plot the results from some of these files.

7.3 Results

Here we will present the results of our evaluation. We will provide a comments for each result to summarize for the reader what the result actually shows. In sub section 7.5 we will provide a summary of all the results and problems in our scenario.

The results are organized as follows. First we will concern the reliability of communication between IVAs, which had an effect on the rest of the analysis. Then we will concern feeling end value results. After that we dig into sub-scenes occurrences and feeling development in the sub-scenes. Afterwards we analyze feeling developments across whole scenario and impact of emotion events on the feeling value. Next we will concern action occurrences and actions and sub-scenes sequencing. Lastly we will outline the sub-scene duration results.

7.3.1 The reliability of communication between IVAs

Here we will plot the statistics of how many messages containing certain action, proposal or text communication were actually lost between the agents. The message loss was caused by an issue hidden in Pogamut 3 core (not by project Emohawk implementation). This issue will be repaired in future Pogamut 3 versions.

Action/prop./conv.	Message loss	Message count
Action bye	525 (15%)	3481
Action compliment	1725 (11%)	15644
Action insult	79 (13%)	605
Action kiss	222 (12%)	1806
Action leave	92 (14%)	651
Action slap	83 (15%)	550
Action sex	196 (13%)	1525

Conversation messages	9779 (23%)	42079
All of the proposals	573 (10%)	5621

Table 4. Missed messages count. The table plots how many of the messages containing actions, proposals and conversations between agents were lost. We have only included actions, proposals and conversation that occurred between three main characters – Anne, Bruno and Clementine.

The message loss is our scenario is indeed a serious issue. Surprisingly, it had only limited negative impact on the overall scenario performance.

Firstly, since our agents reactive system is robust, it can deal even with other agents not responding to their proposals – they will simply repeat them after some time (but only if the feeling value is still high enough).

Secondly, missed messages and actions made the agent interactions even more emergent. So to say, this issue helped our scenario to produce more kinds of agent to agent interactions (e.g. missed proposal may increase the duration of agent to agent interaction). In some cases the missed messages may surely produce non-believable reactions – e.g. if our agent miss negative action towards him, he will not react with negative feeling shift and will not be angry towards the one who performed the action. However this is compensated at the same time. Let's say our agent missed the negative action and is not angry. This means he will not have any desires to depart from the agent that performed the negative action. This can result in the other agent performing another negative action toward our agent – and this second action will be most probably received, causing the desired negative shift. Why would the other agent perform second negative action? Well, if his feeling attitude toward our agent was negative few seconds ago, it will be probably still negative, still allowing for the negative action to be selected as the next action.

Thirdly, regarding the results in Table 4 we can see that average action loss is between 11% and 15%. For proposals it is only 10%. These numbers still allow a reasonable agent to agent interaction with more noise added. Considering the result of conversation messages where more than 20% were lost, we can say that this is actually not a problem at all. The conversation messages have just limited effect upon the emotions – the emotion intensities caused by conversation messages are low compared to emotion intensities caused by actions and proposals.

7.3.2 The feeling end values

We have organized results in this sub section into tables showing feeling relations between the characters Bruno, Anne and Clementine at the end of experiment. The emohawk was excluded from the analysis as we are currently interested more in human to human relations.

Each value in the table shows number of experiments that satisfy conditions given by the row and column headings. We have "lovers" conditions – feeling value of first character towards second character is above two, "friends" conditions – feeling value above one, positive conditions – feeling value above zero and "hate" conditions – feeling value below minus one. Table 5 shows and explains used conditions.

BA>2	Feeling of Bruno to Anne > 2	**AC+0**	Feeling of Anne to Clementine > 0
BC>2	Feeling of Bruno to Clementine > 2	**CA+0**	Feeling of Clementine to Anne > 0
AB>2	Feeling of Anne to Bruno > 2	**BA-0**	Feeling of Bruno to Anne <= 0
CB>2	Feeling of Clementine to Bruno > 2	**BC-0**	Feeling of Bruno to Clementine <= 0
BA>1	Feeling of Bruno to Anne > 1	**AB-0**	Feeling of Anne to Bruno <= 0
BC>1	Feeling of Bruno to Clementine > 1	**CB-0**	Feeling of Clementine to Bruno <= 0
AB>1	Feeling of Anne to Bruno > 1	**AC-0**	Feeling of Anne to Clementine <= 0
CB>1	Feeling of Clementine to Bruno > 1	**CA-0**	Feeling of Clementine to Anne <= 0
AC>1	Feeling of Anne to Clementine > 1	**BA-1**	Feeling of Bruno to Anne < -1
CA>1	Feeling of Clementine to Anne > 1	**BC-1**	Feeling of Bruno to Clementine < -1
BA+0	Feeling of Bruno to Anne > 0	**AB-1**	Feeling of Anne to Bruno < -1
BC+0	Feeling of Bruno to Clementine > 0	**CB-1**	Feeling of Clementine to Bruno < -1
AB+0	Feeling of Anne to Bruno > 0	**AC-1**	Feeling of Anne to Clementine < -1
CB+0	Feeling of Clementine to Bruno > 0	**CA-1**	Feeling of Clementine to Anne < -1

Table 5. Conditions used for feeling end values evaluation. Letters in conditions are the first letters of names of characters the condition is for. For example condition BA means feeling of Bruno to Anne (CB – Clementine to Bruno etc.). Numbers represent the feeling equation the scenario must satisfy in order for the condition to be true. >2 is the best "lovers" condition (feeling value above 2), >1 is second best "friends" condition (feeling above 1 – note that scenario satisfying condition >2 will also satisfy condition >1), number +0 means "positive" condition (feeling value above 0), number -0 means "negative" condition (feeling value below 0) and number -1 is worst "hate" condition (feeling value below -1). Some other conditions may be used (e.g. BC<1 – feeling of Bruno to Anne below 1).

Summarizing Table 5 – the condition sequence from "best" condition to "worst" condition looks as follows:

– BA>2 – Bruno loves Anne – highest measured affect (feeling value > 2), if satisfied BA>1 and BA+0 will be satisfied as well

– BA>1 – Bruno is at least friendly with Anne (feeling value > 1). Bruno may be also in love – all experiments satisfying BA>2 will also satisfy BA>1. If BA>1 satisfied BA+0 will be satisfied as well.

– BA+0 – Bruno is at least positive with Anne (feeling value > 0)

– BA-0 – Bruno is negative with Anne – first negative condition (feeling value < 0)

– BA-1 – Bruno hates Anne – lowest measured affect (feeling value < -1), if satisfied BA-0 will be satisfied as well

Below we will present seven tables containing the results. After each table we will post a short comment what the table actually shows and what are the interesting trends. First two tables 6 and 7 will be concerned with positive relations at the end of scenario between characters. Next three tables 8, 9 and 10 will present an overview of negative relations in the experiments. Last two tables will be concerned with asymmetric relations at the end of experiment.

	BA>2	BC>2	AB>2	CB>2	all
BA>2	**250 (70%)**	89 (25%)	231 (65%)	73 (20%)	-
BC>2	89 (25%)	**179 (50%)**	86 (24%)	158 (44%)	-
AB>2	231 (65%)	86 (24%)	**240 (67%)**	73 (20%)	-
CB>2	73 (20%)	158 (44%)	73 (20%)	**163 (46%)**	-
all	-	-	-	-	**67 (19%)**

Table 6. "Lovers" conditions result. Here we observe number of experiments that ended with someone in love with someone other. Each cell represents the result of logical and applied between condition specified by cell row and condition specified by cell column (e.g. Cell with row BA>2 and column BC>2 shows the count of experiments satisfying BA>2 and BC>2). As we can see more experiments runs ended with Bruno in love with Anne than Bruno in love with Clementine. Also it is worth noting, that the relations are not always symmetric (Bruno loves Anne in 250 experiments, but from these experiments Anne loves Bruno back just in 231 cases). The last row and column in the table marked "all" shows number of experiments where Bruno was in love with both girls and both girls were in love with Bruno (BA>2 and BC>2 and AB>2 and CB>2).

Table 6 shows that there was a strong tendency for the experiments to end with at least two of the characters (boy and one of the girls) with feeling values above two toward each other. Feeling value above two is concerned to represent very positive feeling tendency, because the characters may choose and accept the sex proposal at this time. Also we can observe that these relations tend to be symmetric – Bruno loves Anne in 250 experiments, from these very same experiments Anne loves Bruno back in 231 cases. Anne seems to be Bruno's favorite girl – the reason may be that the Bruno starts the experiment with Anne and he meets Clementine later on. Just in 15 experiments Bruno was not in love with any girl at the end of the experiment (BA<2 and BC<2 = 15 – not shown in the table). Other important values not shown in the table:

– Bruno and Anne in love and Bruno not in love with Clementine – 149 cases (BA>2 and AB>2 and BC<2)

– Bruno and Clementine in love and Bruno not in love with Anne – 86 cases (BA>2 and CB>2 and BA<2)

67 experiments (19%) ended with Bruno in love with both girls and both girls in love with Bruno. This can be caused either by a) conflict less experiment – girls never met each other or b) feeling recovery – a conflict occurred, but later in the scenario Bruno was able to win the girl back or c) a conflict occurred but just between the girls (feeling attitude to Bruno was not hurt that much). Especially b) and c) may occur in the same experiment.

Concerning the emergent performance and believability of the experiments there are several things to note. The experiments tend to end with two of characters in love. However if they will be Clementine and Bruno or Anne and Bruno, that is emergent. Moreover in 67 experiments both of these (symmetric) relations occurred at once. As the relations between characters shape the agent to agent interaction very much, we can proclaim this a success – our experiments produce emergent endings concerning the agent to agent relations.

Concerning the believability the problem here may be asymmetric relations between characters – we will deal with this problem also in the tables later on.

	BA>1	BC>1	AB>1	CB>1	all
BA>1	**268 (75%)**	110 (31%)	262 (73%)	94 (26%)	-
BC>1	110 (31%)	**193 (54%)**	107 (30%)	173 (48%)	-
AB>1	262 (73%)	107 (30%)	**267 (75%)**	93 (26%)	-
CB>1	94 (26%)	173 (48%)	93 (26%)	**179 (50%)**	-
all	-	-	-	-	**86 (24%)**

Table 7. "Friends" conditions result. Here we observe number of experiments that ended at least with someone being a friend with someone other. The numbers are higher here than in Table 6, because the conditions are weaker (experiments satisfying BA>2 will also satisfy BA>1). Anne again beats Clementine as a more popular girl for Bruno. Last row and column again shows number of experiments where symmetric being a friend relation occurred between Bruno and both of the girls.

Table 7 is very similar to Table 6 with the exception that the conditions are now weaker – "friends" conditions require feeling value to be above one. As we would expect the numbers in the table are higher because of this. The relations between Bruno and Anne seem to be slightly more symmetric than the relations between Bruno and Clementine. Even more interesting is the number of experiments Bruno's feeling was below one for both of the girls (BA<1 and BC<1 = 4 – not shown in the table). What happened in these experiments that these conditions occurred? We can see that even simple end feeling value analysis may pinpoint some unexpected experiments endings that are worth more detailed investigation. We will deal with more of these suspicious experiments in the tables below.

	AB+0	CB+0	BA-0	BC-0	BA-1	BC-1
AB+0	**274 (77%)**	121 (34%)	4 (1%)	121 (34%)	1	33 (9%)
CB+0	121 (34%)	**201 (56%)**	58 (16%)	14 (4%)	8 (2%)	4 (1%)
BA-0	4 (1%)	58 (16%)	**58 (16%)**	1	8 (2%)	0
BC-0	121 (34%)	14 (4%)	1	**122 (34%)**	0	33 (9%)
BA-1	1	8 (2%)	8 (2%)	0	**8 (2%)**	0
BC-1	33 (9%)	4 (1%)	0	33 (9%)	0	**33 (9%)**

Table 8. Boy to girl negative relations result. In this table we focused more on the negative relations between the characters – especially Bruno negative relations with the two girls. This table shows some interesting trends as well as some rare results. Some cells in the table are redundant (e.g. BA-0 and BA-1 – all the experiments satisfying BA-1 will also satisfy BA-0).

As the first, Table 8 shows negative relations at the end of experiment. Especially this table helps us to investigate the special cases where Bruno dislikes or even hates one or both of the girls. Moreover the strong asymmetric relations are grasped here as well. Bruno tends to experience negative relation more with Clementine than with Anne with 122 experiments with Bruno with feeling below zero to Clementine against 58 experiments with Bruno with feeling below zero to Anne. Also the extreme hate relation is experienced more toward Clementine (33 to Clementine against 8 to Anne). This is congruent with the tables above (showing Bruno has more positive tendencies to Anne).

Furthermore this table supports our observation that our experiments tend to end with Bruno experiencing positive feeling relation with at least one of the girls. There were no experiments in which Bruno would hate both of the girls – even in which Bruno would hate just one of the girls and with second he would be just below zero. The rare special cases worth detailed analysis are here:

- Bruno below zero with both of the girls – one experiment
- Anne positive towards Bruno but Bruno hates Anne – one experiment
- Clementine positive towards Bruno but Bruno hates Clementine – four experiments

	AC+0	CA+0	AC-0	CA-0	AC-1	CA-1
AC+0	**9 (3%)**	2	-	7 (2%)	-	4 (1%)
CA+0	2	**7 (2%)**	5 (1%)	-	2	-
AC-0	-	5 (1%)	**346 (96%)**	341 (96%)	-	185 (52%)
CA-0	7 (2%)	-	341 (96%)	**348 (97%)**	184 (52%)	-
AC-1	-	2	-	184 (52%)	**186 (52%)**	111 (31%)
CA-1	4 (1%)	-	185 (52%)	-	111 (31%)	**189 (53%)**

Table 9. Girl to girl relations result. This table focuses on the end relation between the two girls in the 357 experiments. As we can see the girls are more enemies than friends. It is worth noting that no experiment fulfilled conditions AC>1 or CA>1 (not shown in this table).

We have now some idea how the relations between Bruno and the two girls are – but what about girl to girl relations? Table 9 shows results in this matter. We can observe that girl to girl relations tend to end negative – more than 95% of experiments ended with both girls being negative toward each other, 31% of experiments with both girls hating each other. This is due to the nature of the scenario, where Bruno performs actions toward a girl that are perceived negatively by the other girl. Moreover the girls are programmed in a way that they will never stop and chat with each other when they meet. Only way to achieve girl to girl interaction is through Bruno, which will obviously have high probability of causing conflicts. The rare cases where girls actually ended with positive feeling toward each other are worth detailed investigation. But the reason will be they probably have not met at all in the scenario.

	BA>1	BC>1	AB-1	AC-1	CB-1	CA-1	AB-1, AC-1	AB-0 AC-0	CB-1, CA-1	CB-0 CA-0
BA>1	**268 (75%)**	110	2	140	52	118	0	4	32	147
BC>1	110	**193 (54%)**	32	69	4	102	18	79	2	13
AB-1	2	32	**32 (9%)**	18	0	27	-	32	0	0
AC-1	140	69	18	**186 (52%)**	38	18	-	42	25	111
CB-1	52	4	0	38	**52 (16%)**	32	0	0	-	50
CA-1	118	102	27	18	32	**189 (53%)**	14	66	-	84
AB-1, AC-1	0	18	-	-	0	14	**18 (5%)**	-	0	0
AB-0 AC-0	4	79	32	42	0	66	-	**81 (23%)**	0	1
CB-1, CA-1	32	2	0	25	-	-	0	0	**32 (9%)**	-
CB-0 CA-0	147	13	0	111	50	84	0	1	-	**148 (41%)**

Table 10. Girl to girl and girl to boy negative relations. This complex table features above else three conditions matching. Some of the rows and columns feature a pair of condition instead of one. These conditions are evaluated with logical and. Except of this the evaluation proceeds normal – so the pair of condition is evaluated with logical and against the other condition or pair of condition got from row or column. Due to better orientation percents are shown just for the diagonal cells.

Table 10 concerns asymmetric relations between Bruno and the girls again – focusing more on "Bruno likes a girl, but the girl hates him" issue. Some of the numbers here have been already in the tables above. Again the rare, suspicious cases are of the biggest interest for us:

– two experiments ending with Anne hating Bruno, while Bruno was at least a friend with Anne

– four experiments ending with Clementine hating Bruno, while Bruno was at least friendly towards her

Interesting is also an experiment where both of the girls liked no one – they were negative toward each other and also to Bruno. Considering the symmetric "hate" relations between two girls we can see that although these relations are less symmetric, there are other trends such as – when Anne hates Clementine (186 experiments), Clementine is negative toward Bruno and Anne in 111 of these experiments (for Clementine this trend is not that strong). Moreover in all of 32 cases where Clementine "hated" both Anne and Bruno, Bruno was at least friendly with Anne.

The observations are:

1.) The trend is that at least some of the relations between agents tend to be positive – this emerged from the affective-architecture design.

2.) Positive feeling between two characters seems to have impact onto the relation with the third character – if there is stronger or weaker negative relation of third character toward other two characters, these characters feature positive attitude toward each other in 97% of the cases.

	AB-0
BA>1	4 (1.1%)

	BA-0
AB>1	0 (0%)

	AB-1
BA>2	1 (0.3%)

	CB-0
BC>1	13 (3.6%)

	BC-0
CB>1	2 (0.6%)

	CB-1
BC>2	1 (0.3%)

Table 11. Strong asymmetric relations between characters. This table shows the number of experiments that ended with strong asymmetric relations between Bruno and one of the girls.

This multi-table 11 summarizes strong asymmetric relations between characters. All of the experiments here are worth detailed analysis. The worst results are two experiments featuring Bruno in love with the girl, while the girl hates him. Strong asymmetric relations occur in 5.8% of all experiments.

	AB-0
BA+0	27 (8%)

	AB<1
BA>1	6 (1.5%)

	BA-0
AB+0	4 (1.1%)

	CB-0
BC+0	46 (13%)

	CB<1
BC>1	20 (5.6%)

	BC-0
CB+0	14 (4%)

Table 12. Weak asymmetric relations between characters. This table shows the number of experiments that ended with at least weak asymmetric relations between Bruno and one of the girls.

Multi-table 12 summarizes weak asymmetric relations between characters. The higher numbers than in Table 11 are caused by the weaker requirements on the experiments. Weak asymmetric relations occur in approximately 25% of all the experiments.

Summary. In this sub-section we have presented the reader with the first results summarizing the feeling end values of human characters in the scenario toward other human characters in the scenario. Considering scenario emergent performance, there seem to be several possible scenario ends:

– 149 cases (42%) ended with symmetric relation Bruno and Anne in love, but Bruno was not in love with Clementine

– 86 cases (24%) ended with symmetric relation Bruno and Clementine in love but Bruno

was not in love with Anne

– 67 cases (19%) ended with symmetric relation Bruno and Anne in love and also with symmetric relation Bruno and Clementine in love

Considering the girl to girl relations:

– in 111 cases (31%) Anne and Clementine hated each other

– in 230 cases (64%) Anne and Clementine disliked each other, but were not hating each other

Indeed there would be many other possible story endings if we would take into account all possible conditions combination, but to illustrate our scenario produces emergent story endings, we believe this is enough.

Other question is how many experiments featured suspicious relations at the end. Or in other words – how many experiments we should investigate in greater detail? Considering strong asymmetric relations we have 21 (5.8%) experiments to investigate. This is a good result for our prototype implementation. Weaker asymmetric relations occur more, as would one expect. We should investigate at least some of them to find if there are some trends or other conditions that has to be fulfilled in order for weak asymmetry to occur. Also the special end conditions experiments are worth investigation (e.g. the experiment where Bruno dislikes both of the girls). Considering the other experiments, we will analyze them with more methods to find out which of them are worth further investigation.

7.3.3 The sub-scenes occurrences

Here we will analyze our experiments using the sub-scenes occurrences. We will start with plotting simple sub-scenes counts for all of the characters (sub-scenes listed alphabetically). Then we will look on the key point sub-scene occurrence in the experiments.

Before we will present the results we would like to say a few words on how to "read" sub-scenes identifiers.

– AGENT_ALONE – this indicates our agent was completely alone at the time. Although there may be some agents around, none of them is selected as "agent with".

– INTERRUPTED – this indicates our agent was in interrupted state. Some other agent or agents have approached him and talked to him.

– WITH_BRUNO, WITH_ANNE, WITH_CLEMENTINE – indicates the name of the agent that is currently selected as an "agent with" by our agent.

– AT_ or ON_WAY_TO_ – each identifier ends either with AT and some string or ON_WAY_TO_ and some string. AT indicates the place our agent was at (e.g. park, cinema, girls home, etc.). ON_WAY_TO indicates that our agent is currently going toward some goal place in the environment.

– APPROACH_BOYS – this means the girl is currently in approach boys state, where she is actively exploring the environment and interacting with any male characters she may encounter.

There identifier strings are combined together to create complete sub-scenes identifier. E.g. The full identifier "INTERRUPTED_WITH_BRUNO_ON_WAY_TO_MY_HOME " can be read as *"I was going home with Bruno, when suddenly someone required our attention."*.

Anne sub-scene counts. Here we will plot sub-scenes counts for Anne obtained from 357 experiments. During these 357 experiments Anne was in 4783 sub-scenes. That is roughly 13 sub-scenes for one experiment. Note that Anne can be in two same sub-scenes in one experiment (e.g. Bruno may return to Anne later in the experiment and take her to cinema and then home again).

- AGENT_ALONE_AT_ANNE_HOME – count – 816 (~ 2,3 per experiment)
- AGENT_ALONE_AT_CINEMA – count – 5
- AGENT_ALONE_AT_CLEMENTINE_HOME – count – 1
- AGENT_ALONE_AT_ELSEWHERE – count – 347 (~ 1 per experiment)
- APPROACH_BOYS_AT_CINEMA – count – 1
- APPROACH_BOYS_AT_ELSEWHERE – count – 798 (~ 2,2 per experiment)
- INTERRUPTED_ALONE_AT_ANNE_HOME – count – 23
- INTERRUPTED_ALONE_AT_CINEMA – count – 22
- INTERRUPTED_ALONE_AT_ELSEWHERE – count – 93
- INTERRUPTED_WITH_BRUNO_AT_ANNE_HOME – count – 8
- INTERRUPTED_WITH_BRUNO_AT_CINEMA – count – 177 (~ 0,5 per experiment)
- INTERRUPTED_WITH_BRUNO_AT_CLEMENTINE_HOME – count – 3
- INTERRUPTED_WITH_BRUNO_AT_ELSEWHERE – count – 98
- INTERRUPTED_WITH_BRUNO_ON_WAY_TO_CINEMA – count – 5
- INTERRUPTED_WITH_BRUNO_ON_WAY_TO_MY_HOME – count – 118
- WITH_BRUNO_AT_ANNE_HOME – count – 481 (~ 1,3 per experiment)
- WITH_BRUNO_AT_CINEMA – count – 352 (~ 1 per experiment)
- WITH_BRUNO_AT_CLEMENTINE_HOME – count – 4
- WITH_BRUNO_AT_ELSEWHERE – count – 472 (~ 1,3 per experiment)
- WITH_BRUNO_ON_WAY_TO_CINEMA – count – 219 (~ 0,6 per experiment)
- WITH_BRUNO_ON_WAY_TO_MY_HOME – count – 726 (~ 2 per experiment)
- WITH_BRUNO_ON_WAY_TO_PARK – count – 14

Considering Anne sub-scenes results we should focus on all the sub-scenes with low count of occurrence. Some of them show some undesirable features of our scenario. For example – all Anne's sub-scenes with "AT_CLEMENTINE_HOME" should not occur in our scenario (girls should not be allowed to freely visit the other girl home). We can prevent this by denying girls to visit each other home. However, this is the only problem here. Anne experienced 547 interrupted sub-scenes (possibly featuring some conflict) against 4236 other non-conflict sub-scenes.

Bruno sub-scene counts. Here we will plot sub-scenes counts for Bruno obtained from 357 experiments. During these 357 experiments Bruno was in 6609 sub-scenes. That is roughly 19 sub-scenes for one experiment. Reason for Bruno being in more sub-scenes than Anne is Bruno can experience higher number of sub-scenes (e.g. Bruno can be at cinema both "with Anne" and "with Clementine", Anne can be at cinema just "with Bruno").

- AGENT_ALONE_AT_ANNE_HOME – count – 221 (~ 0,6 per experiment)
- AGENT_ALONE_AT_CINEMA – count – 1
- AGENT_ALONE_AT_CLEMENTINE_HOME – count – 205 (~ 0,6 per experiment)
- AGENT_ALONE_AT_ELSEWHERE – count – 1142 (~ 3,2 per experiment)
- INTERRUPTED_ALONE_AT_ANNE_HOME – count – 46
- INTERRUPTED_ALONE_AT_CINEMA – count – 3
- INTERRUPTED_ALONE_AT_CLEMENTINE_HOME – count – 22

- INTERRUPTED_ALONE_AT_ELSEWHERE – count – 31
- INTERRUPTED_WITH_ANNE_AT_ANNE_HOME – count – 9
- INTERRUPTED_WITH_ANNE_AT_CINEMA – count – 56
- INTERRUPTED_WITH_ANNE_AT_CLEMENTINE_HOME – count – 2
- INTERRUPTED_WITH_ANNE_AT_ELSEWHERE – count – 6
- INTERRUPTED_WITH_ANNE_ON_WAY_TO_CINEMA – count – 7
- INTERRUPTED_WITH_ANNE_ON_WAY_TO_GIRL1_HOME – count – 70
- INTERRUPTED_WITH_CLEMENTINE_AT_ANNE_HOME – count – 3
- INTERRUPTED_WITH_CLEMENTINE_AT_CINEMA – count – 130
- INTERRUPTED_WITH_CLEMENTINE_AT_CLEMENTINE_HOME – count – 62
- INTERRUPTED_WITH_CLEMENTINE_AT_ELSEWHERE – count – 9
- INTERRUPTED_WITH_CLEMENTINE_ON_WAY_TO_CINEMA – count – 38
- INTERRUPTED_WITH_CLEMENTINE_ON_WAY_TO_GIRL2_HOME – count – 14
- INTERRUPTED_WITH_CLEMENTINE_ON_WAY_TO_PARK – count – 2
- WITH_ANNE_AT_ANNE_HOME – count – 485 (~ 1,3 per experiment)
- WITH_ANNE_AT_CINEMA – count – 295 (~ 0,8 per experiment)
- WITH_ANNE_AT_CLEMENTINE_HOME – count – 19
- WITH_ANNE_AT_ELSEWHERE – count – 609 (~ 1,7 per experiment)
- WITH_ANNE_AT_PARK – count – 1
- WITH_ANNE_ON_WAY_TO_CINEMA – count – 208 (~ 0,6 per experiment)
- WITH_ANNE_ON_WAY_TO_GIRL1_HOME – count – 690 (~ 1,9 per experiment)
- WITH_ANNE_ON_WAY_TO_PARK – count – 14
- WITH_CLEMENTINE_AT_ANNE_HOME – count – 14
- WITH_CLEMENTINE_AT_CINEMA – count – 373 (~ 1 per experiment)
- WITH_CLEMENTINE_AT_CLEMENTINE_HOME – count – 428 (~ 1,2 per experiment)
- WITH_CLEMENTINE_AT_ELSEWHERE – count – 506 (~ 1,4 per experiment)
- WITH_CLEMENTINE_AT_PARK – count – 3
- WITH_CLEMENTINE_ON_WAY_TO_CINEMA – count – 371 (~ 1 per experiment)
- WITH_CLEMENTINE_ON_WAY_TO_GIRL2_HOME – count – 495 (~ 1,4 per experiment)
- WITH_CLEMENTINE_ON_WAY_TO_PARK – count – 29

The problem of a girl being at home of another girl can be clearly visible as well in Bruno results. 33 of these sub-scenes occurred in our experiments for Bruno. Considering the somewhat weird result that Bruno was more times at cinema with Clementine than with Anne we have to say, that this is due to the fact, that although Bruno and Anne starts the scenario "going out" of the cinema, they are not exactly at cinema position. Due to this fact, first sub-scene for them is "WITH_ANNE_AT_ELSEWHERE" and "WITH_BRUNO_AT_ELSEWHERE". This is the explication of suspicious result with Bruno being at cinema more with Clementine than with Anne.

Clementine sub-scene counts. Here we will plot sub-scenes counts for Clementine obtained from 357 experiments. During these 357 experiments Clementine was in 4493 sub-scenes. That is roughly 13 sub-scenes for one experiment. These numbers resembles Anne's numbers.

- AGENT_ALONE_AT_CINEMA – count – 4
- AGENT_ALONE_AT_CLEMENTINE_HOME – count – 506 (~ 1,4 per experiment)
- AGENT_ALONE_AT_ELSEWHERE – count – 295 (~ 0,8 per experiment)

- APPROACH_BOYS_AT_CINEMA – count – 2
- APPROACH_BOYS_AT_ELSEWHERE – count – 641 (~ 1,8 per experiment)
- INTERRUPTED_ALONE_AT_CINEMA – count – 25
- INTERRUPTED_ALONE_AT_CLEMENTINE_HOME – count – 48
- INTERRUPTED_ALONE_AT_ELSEWHERE – count – 66
- INTERRUPTED_WITH_BRUNO_AT_ANNE_HOME – count – 2
- INTERRUPTED_WITH_BRUNO_AT_CINEMA – count – 147 (~ 0,4 per experiment)
- INTERRUPTED_WITH_BRUNO_AT_CLEMENTINE_HOME – count – 64
- INTERRUPTED_WITH_BRUNO_AT_ELSEWHERE – count – 74
- INTERRUPTED_WITH_BRUNO_ON_WAY_TO_CINEMA – count – 43
- INTERRUPTED_WITH_BRUNO_ON_WAY_TO_MY_HOME – count – 30
- INTERRUPTED_WITH_BRUNO_ON_WAY_TO_PARK – count – 2
- WITH_BRUNO_AT_ANNE_HOME – count – 3
- WITH_BRUNO_AT_CINEMA – count – 395 (~ 1,1 per experiment)
- WITH_BRUNO_AT_CLEMENTINE_HOME – count – 740 (~ 2,1 per experiment)
- WITH_BRUNO_AT_ELSEWHERE – count – 476 (~ 1,3 per experiment)
- WITH_BRUNO_AT_PARK – count – 3
- WITH_BRUNO_ON_WAY_TO_CINEMA – count – 386 (~ 1,1 per experiment)
- WITH_BRUNO_ON_WAY_TO_MY_HOME – count – 512 (~ 1,4 per experiment)
- WITH_BRUNO_ON_WAY_TO_PARK – count – 29

Clementine sub-scenes show again the problem with Clementine being at Anne's home. The reason why sub-scene "ON_WAY_TO_CINEMA" occurred just 43 times and sub-scene "WITH_BRUNO_AT_CINEMA" 395 is that Bruno usually takes Clementine to cinema from the meeting place that is actually very close to cinema. Due to this, characters can make it from there to cinema in less than five seconds, which is the minimal duration for sub-scene to be generated. Also Clementine may encounter Bruno at cinema with Anne, when she was exploring the environment in state "approach boys", shifting her potentially to state "WITH_BRUNO_AT_CINEMA" before she will be interrupted by Anne.

Sub-scenes key points. Here we will perform analysis of the experiments regarding occurrence of certain sub-scenes for certain characters in experiments. We will consider Bruno and "WITH_CLEMENTINE" sub-scenes as we found out, that in small number of experiments Bruno was not with Clementine at all. Also we will review all the experiments, where Bruno was not with Clementine at her home – sub-scene "WITH_CLEMENTINE_ AT_CLEMENTINES_HOME" missing.

Not "WITH_CLEMENTINE". In 1.1% of all the experiments Bruno did not experience any sub-scene "WITH_CLEMENTINE". By reviewing the experiments we have found out the reason is as follows: When Bruno is with Anne and performs positive actions, Clementine observes some of them, which causes negative feeling shift towards Bruno. Then when Clementine interrupts Bruno with Anne, she performs negative actions to Bruno that cause negative feeling shift for Bruno's attitude to Clementine. From now on, Bruno has no desire to speak to Clementine again. For these conditions to occur in the experiment, it is needed that Bruno and Anne at the beginning of the experiment are "slow" – they take long time to agree to go to Anne's home. This can result in a) Bruno will not make it in time to pick up Clementine at cinema and b) Clementine may observe some of the positive actions between Anne and Bruno from a distance.

Note that not "with Clementine" experiments will feature negative end feeling value of

Bruno to Clementine, but not all of the experiments featuring negative end feeling value between Bruno and Clementine will miss "with Clementine" sub-scene. The reason is the critical conflict may occur after Bruno meets Clementine for the first time.

Not "AT_CLEMENTINES_HOME". In 19% of all the experiments Bruno was not in sub-scene "WITH_CLEMENTINE_AT_CLEMENTINES_HOME". We have reviewed all of them and found out following reasons.

In 54% of these experiments following conditions occurred: Scenario goes on normally, but at the time Bruno is with Clementine at cinema, Anne arrives. Conflict occurs and as a result Clementine is angry and walks away. After that Bruno stays with Anne.

In 16% of these experiments again the conflict at cinema occurs, but this time Clementine feeling is not low enough to leave Bruno. She stays with Bruno for some time (after Anne leaves), but since her feeling is below zero some negative actions eventually occurs decreasing Bruno feeling enough to leave Clementine alone.

In 18% of these experiments the conflict at cinema ends by both of the girls walking away. Bruno eventually resumes relations with Anne, because Clementine feeling is usually too low to allow any further positive interaction.

The reasons above are not in conflict with scenario believability. They show the interaction outcomes in the scenario are emergent. The 12% of the experiments featured other reasons to presented behavior (e.g. failure in path-finding code – discussed at the end of this chapter).

Note that not in all experiments, where Clementine experienced negative end feeling value toward Bruno, must be the "at Clementine home" scene missing. Again the conflict may occur after Bruno was at home with Clementine.

7.3.4   The sub-scenes feeling development – a) differences

In this sub section we investigate the feeling development during scenario sub-scenes. We will do so by analyzing feeling differences during each sub-scene. By feeling differences, we mean the difference between the feelings at the beginning of the sub-scene and at the end of the sub-scene for each character toward other characters. This sub-section will present three graphs – each graph will plot box plots showing the feeling differences for all the sub-scenes the character had experienced. In following sub section 7.3.5 we will complete sub-scene feeling development analyze by analyzing feeling values at the end of each sub-scene.

Sometimes in this chapter we will speak about box plots showing feelings increases or decreases, even when there are some values in the box plot that contradicts these trends. Due to the nature of result data, we are interested more in general trends. This means when it can clearly seen that vast majority of values is above zero we will speak about this box plot as a box plot that is showing positive feeling attitude. Data plotted by the graphs below contains a lot of noise – e.g. positive feeling difference does not mean that the actual feeling was above zero (although usually it will be) – maybe just some negative emotion was loosing its intensity and feeling value was going up. This is the reason why we will speak just about general trends here. The next sub section will finish the feeling development analyze clearing some results obtained here.

Note that due to the small number of occurrences of certain sub-scenes, the results for them shown on the graphs are not relevant. On each graph we have selected and commented the most of the relevant results the graph produced. At the end of this sub-section we will provide summary of our findings.
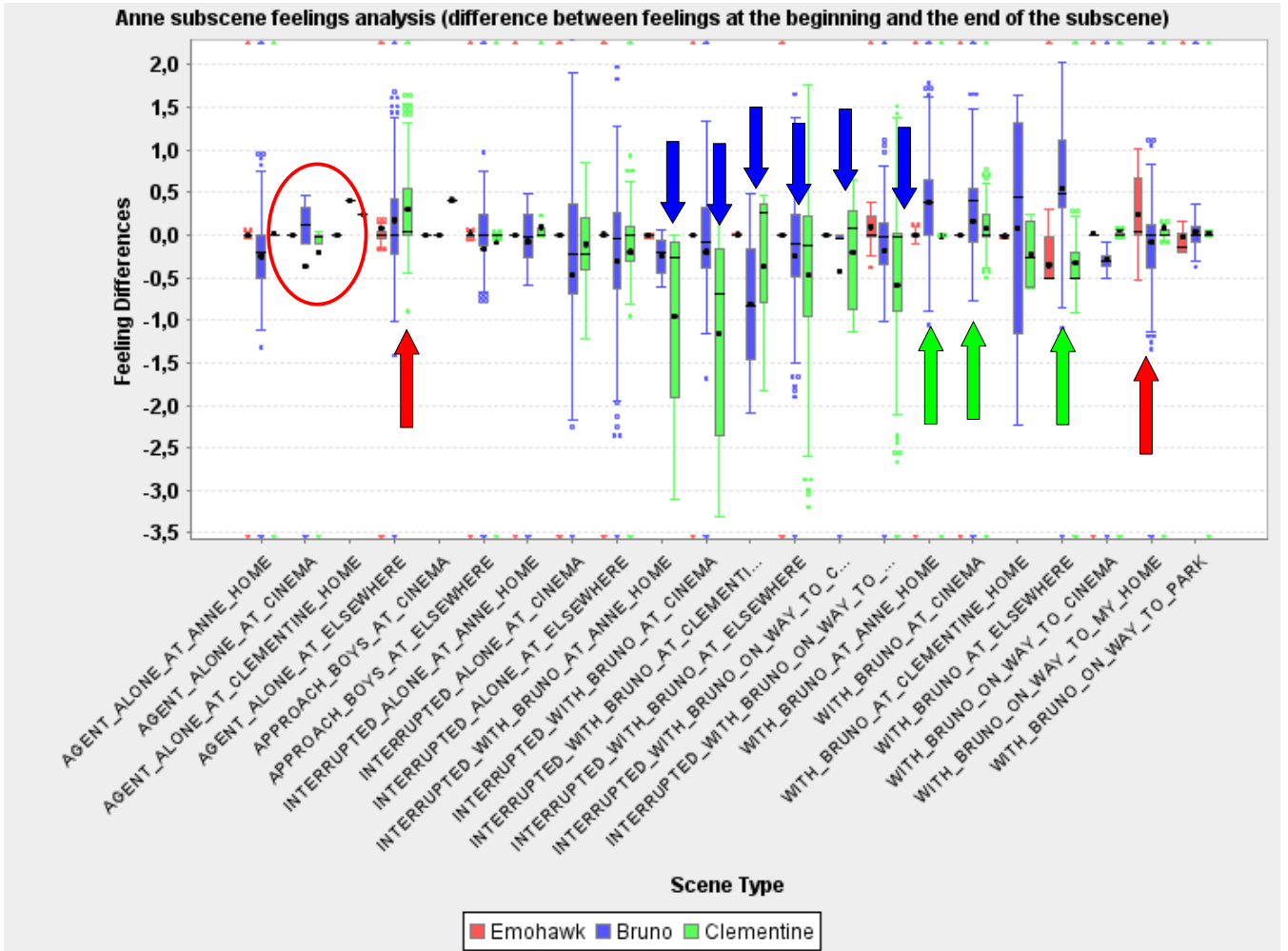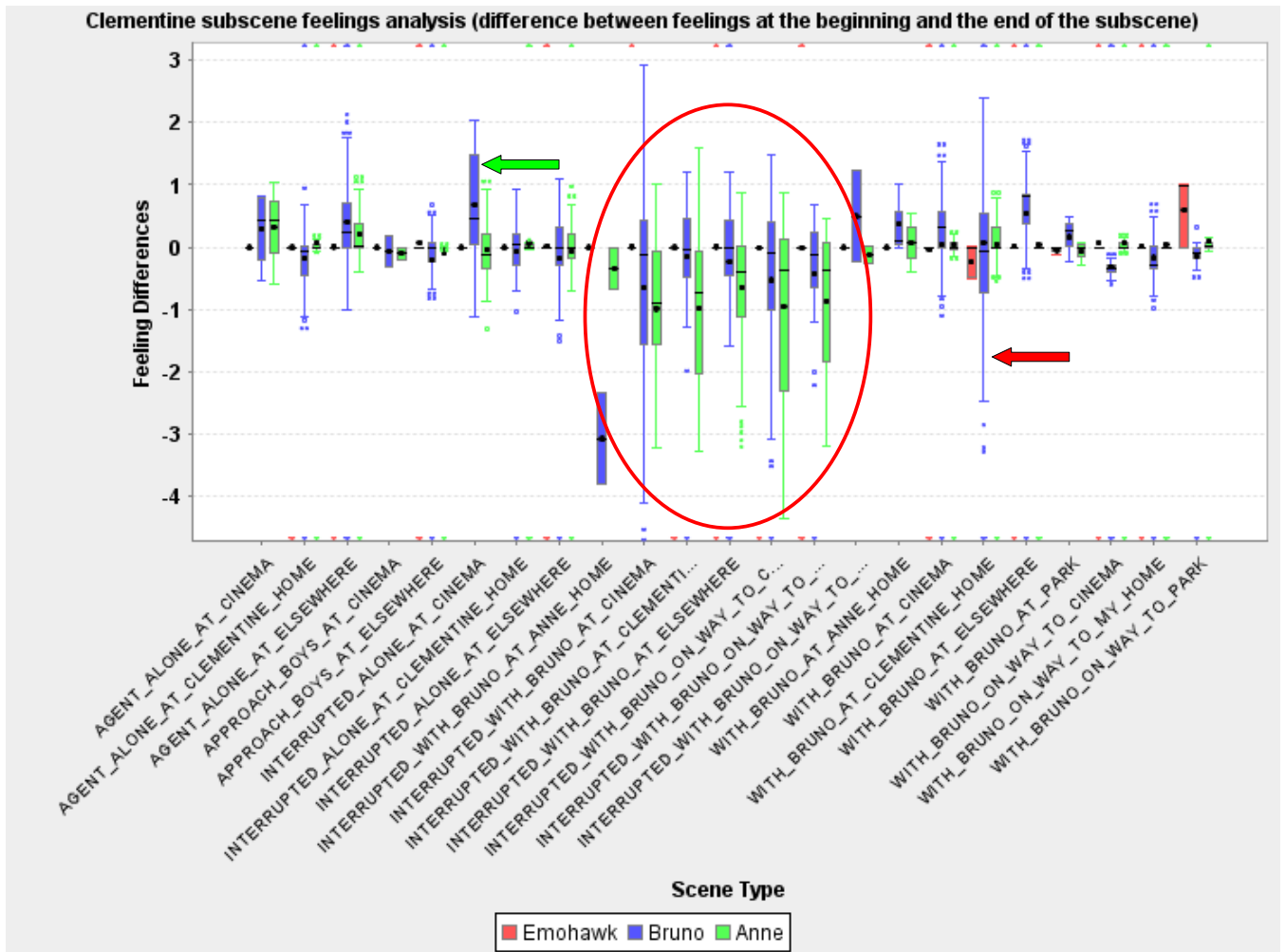
Graph 4. Anne's sub-scene feelings differences. Graph is showing feeling differences during all the sub-scenes Anne had experienced in the experiments. This is the first box plot graph we present, so we will describe what the box plot shows. Each box plot consists of a filled rectangle with two whiskers. The rectangle shows an area where 50% of all the values are situated (values between the first quartile and the third quartile). The whiskers are showing the location of highest and smallest value, that is not considered as an outlier (an outlier is a value that lies more than 1.5 * inter-quartile higher than first quartile or lower than third quartile). The outliers are marked by dots. The median is marked by the vertical black line inside the box plot rectangle. The mean value is marked by large black dot (usually near the median). The arrows and ellipses on the graph will be always commented in the text below.

Graph 4 shows Anne's feeling differences toward other characters in Anne's sub-scenes. First we can observe, that all of the sub-scenes "interrupted with Bruno..." are showing fairly strong feeling decreases toward Clementine (marked by blue arrows). Three green arrows are showing positive feeling increments toward Bruno, when Anne is with him uninterrupted. Blue and green arrows mark desired results with sub-scenes featuring three agents ("interrupted with") showing feeling decreases (due to possible conflicts) and sub-scenes featuring two agents (boy and girl) showing feeling increases.

Sub-scene "with Bruno at Clementine home" is not marked due to just 4 occurrences in the experiments. Red arrow on the left is pointing toward interesting trend – increasing the feeling toward Clementine when Anne is completely alone. In fact this is a believable feature of the model due to the fact, then the strong negative emotions tend to decrease over time and

63

since the Anne is not with Clementine when she is alone, the trend will be as it is.

The right red arrow shows that the feeling shifts connected with emohawk are present for Anne almost exclusively on the way to Anne's home (Bruno and Anne walk around emohawk when going from cinema to Anne's home). It can be clearly seen that the emohawk is friendly toward characters, producing positive feeling shifts. This is a result of emohawk being programmed to be friendly in our scenario prototype.

Note the red ellipse at the left side of the graph. It is showing some of the sub-scenes that have not occurred often enough to produce relevant data (the resulting box plot often miss the rectangular part as well as whiskers).
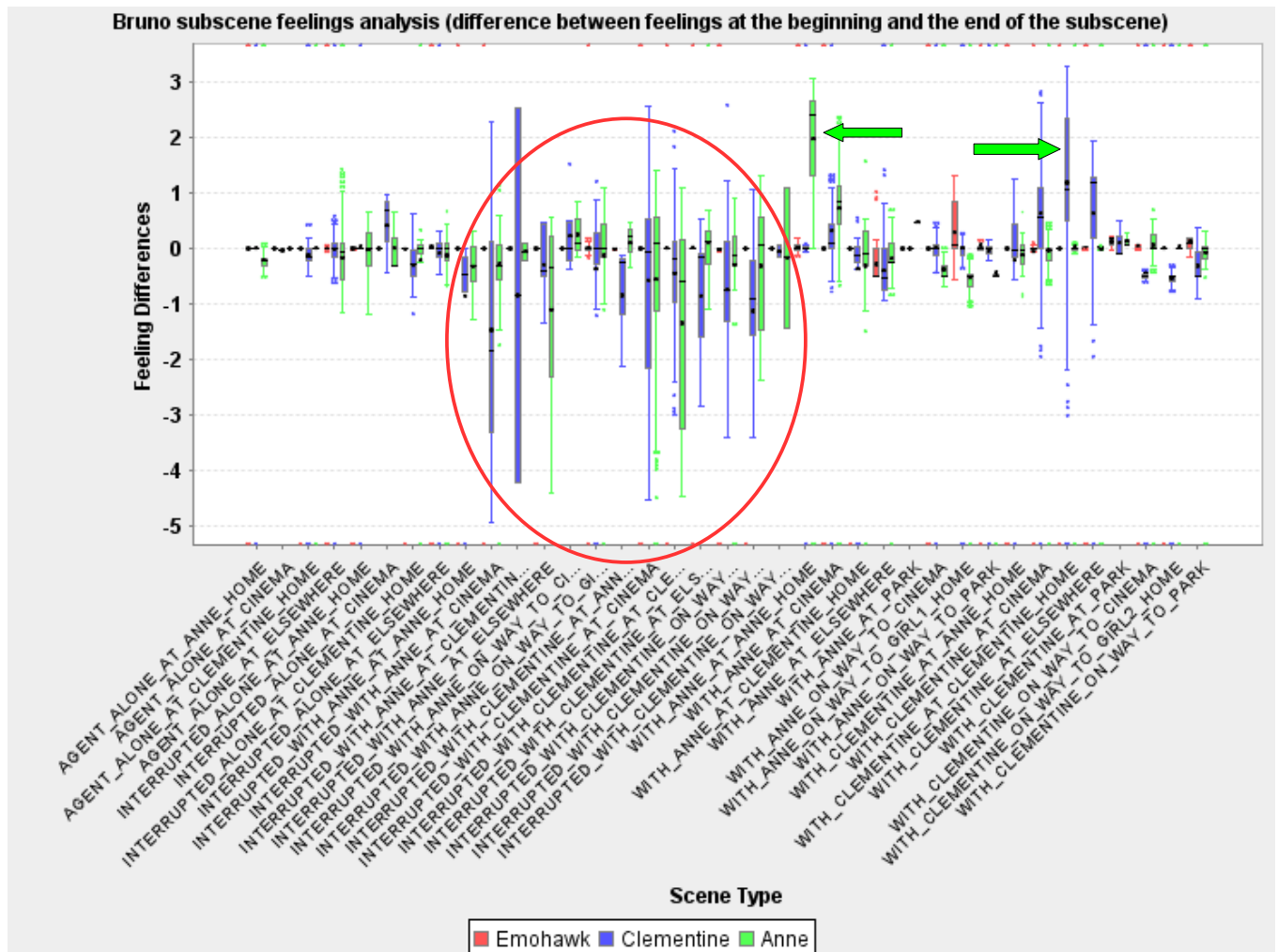
Graph 5. Bruno's sub-scene feelings differences. Graph is showing feeling differences during all the sub-scenes Bruno had experienced in the experiments.

At first glance Bruno's Graph 5 features higher number of sub-scenes than the graph of Anne. This is due to the fact that Bruno can have two characters selected as his "agent with" – Anne and Clementine. All the sub-scenes concerning "agent with" are doubled then. Red ellipse shows the area of "interrupted" sub-scenes. Again the clear trend is the feeling will decrease – toward both of the girls. This is due the conflict occurrence in sub-scenes featuring three human characters. On the other hand, the green arrows are showing nice trend of greater feeling increase when Bruno is with a girl at her home. At home the most positive actions –

kiss and sex may occur having increasing effect on the feeling between two agents.

Graph 6. Clementine's sub-scene feelings differences. Graph is showing feeling differences during all the sub-scenes Clementine had experienced in the experiments.

Clementine's "interrupted..." sub-scenes marked by a red ellipse on Graph 6 continue to show a nice trend of feeling decrease in three characters scenes. On the other hand, there are several others phenomens to note. The red arrow is pointing on sub-scene where Clementine is uninterrupted at her home with Bruno. Here we would certainly expect bigger feeling increase trend. The values are evenly distributed with the mean value being slightly positive. Why the trend here is not positive feeling increase as in the case of Anne? Is this a wrong result? We will see in the next sub-section (considering sub-scenes end feeling values), that it is not as bad as it looks.

The green arrow is showing a sub-scene occurring in 25 cases, where Clementine has no agent selected as "agent with", but is interacting with someone at cinema (either Bruno or Anne). The positive feeling shift to Bruno here may be the result of either a) some negative emotions decaying after currently experienced conflict or b) Bruno performing positive actions to Clementine after she wished to end interaction with him for some reason (hence he is not treated as "agent with" but as an interrupter with positive feeling attitude toward him).

Summary. We have analyzed sub-scenes feeling differences for all of the characters. Note

65

that the feeling differences are just one part of the story. To fully understand the feeling shifts we need to commit further analysis – analysis regarding the feeling values at the end of each sub-scene. We will do so in the next sub section, providing there the reader with complete summary for this and oncoming sub section.

7.3.5 The sub-scenes feeling development – b) end values

Here we will conclude sub-scene feeling development analysis by analyzing feeling values at the end of sub-scenes. This sub section will provide three graphs – one for each character in the scenario – plotting box plots with the feeling values at the end of each sub-scene.
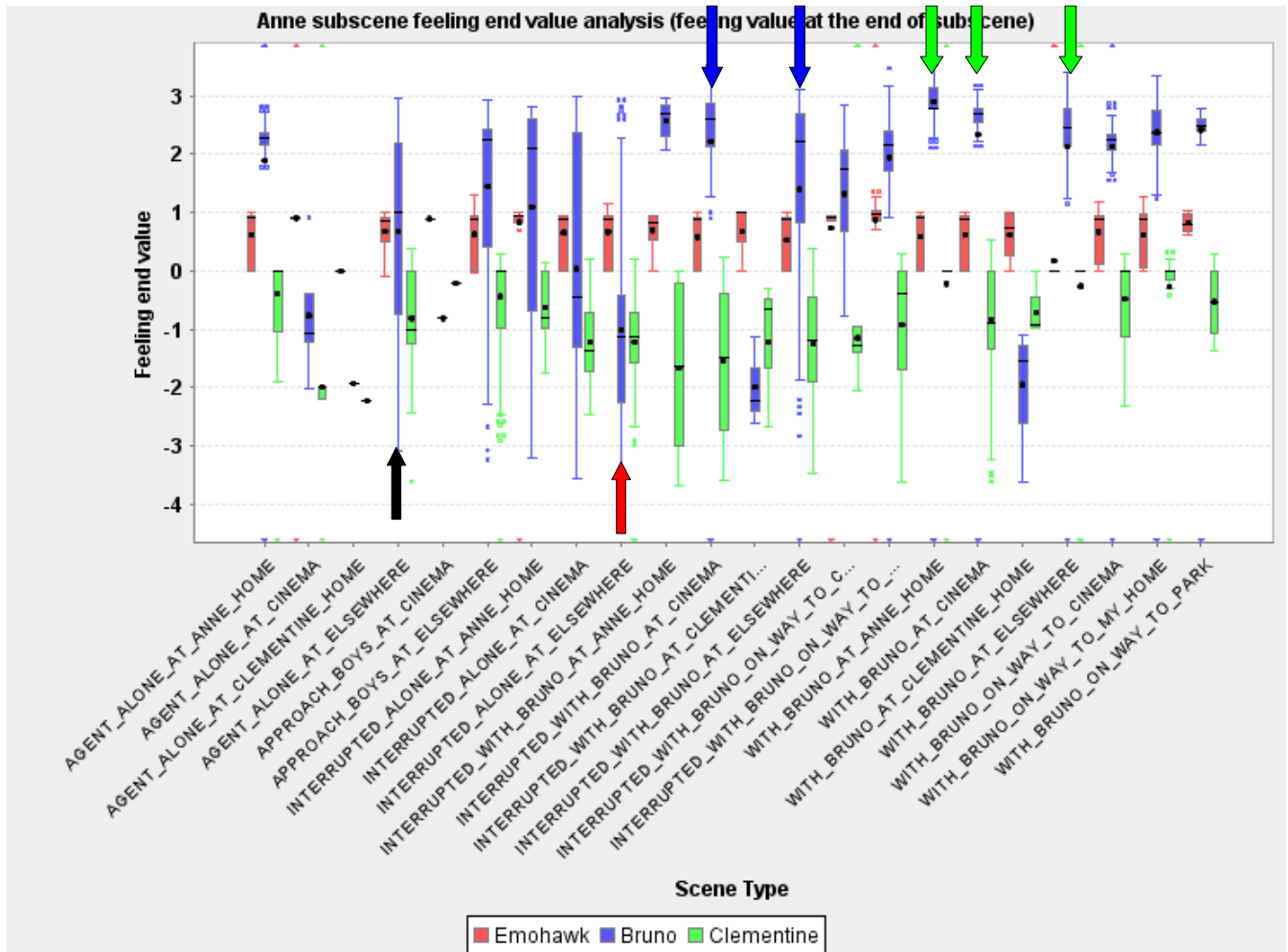
Graph 7. Anne's sub-scene end feeling values. Graph is showing end feeling values for all the sub-scenes Anne had experienced in the experiments.

On Graph 7 three green arrows are marking the results from "with Bruno" sub-scenes. We can see that the value toward Bruno was here positive and above zero. This is desired – in these sub-scenes the characters should like each other.

If we observe feeling values for Clementine we can see that all of the green box plots are below zero, occasionally with few values above it. This is the reason of our girls unable to perform positive actions to each other. Also the conflicts caused by Bruno doing positive actions to a girl seen by the other girl are further lowering the feeling value. We can see the

most negative sub-scenes are interrupted sub-scenes again. The only relevant interrupted sub-scene where Bruno's trend is to end below zero is shown by a red arrow. In the sub-scene it is noted that Anne was "alone" – this means she did not like Bruno enough to make him her "agent with" or recently ended the interaction with him. This is congruent with a majority of feeling values being below zero.

The other "interrupted" sub-scenes marked by blue arrows tend to end for Bruno positively. Bruno is here "agent with" of Anne. Considering the fact the initial feeling value in this sub-scene for Bruno was quite high, this is an understandable trend.

Some of the box plots have values scattered between interval -3 and 3. This shows that the associated sub-scene can occur after various encounters in the scenario. E.g. marked by a black arrow on Graph 7 we observe "agent alone at home" sub-scene. This sub-scene may occur before or after some conflict in the scenario takes place, hence the values will be distributed more evenly.
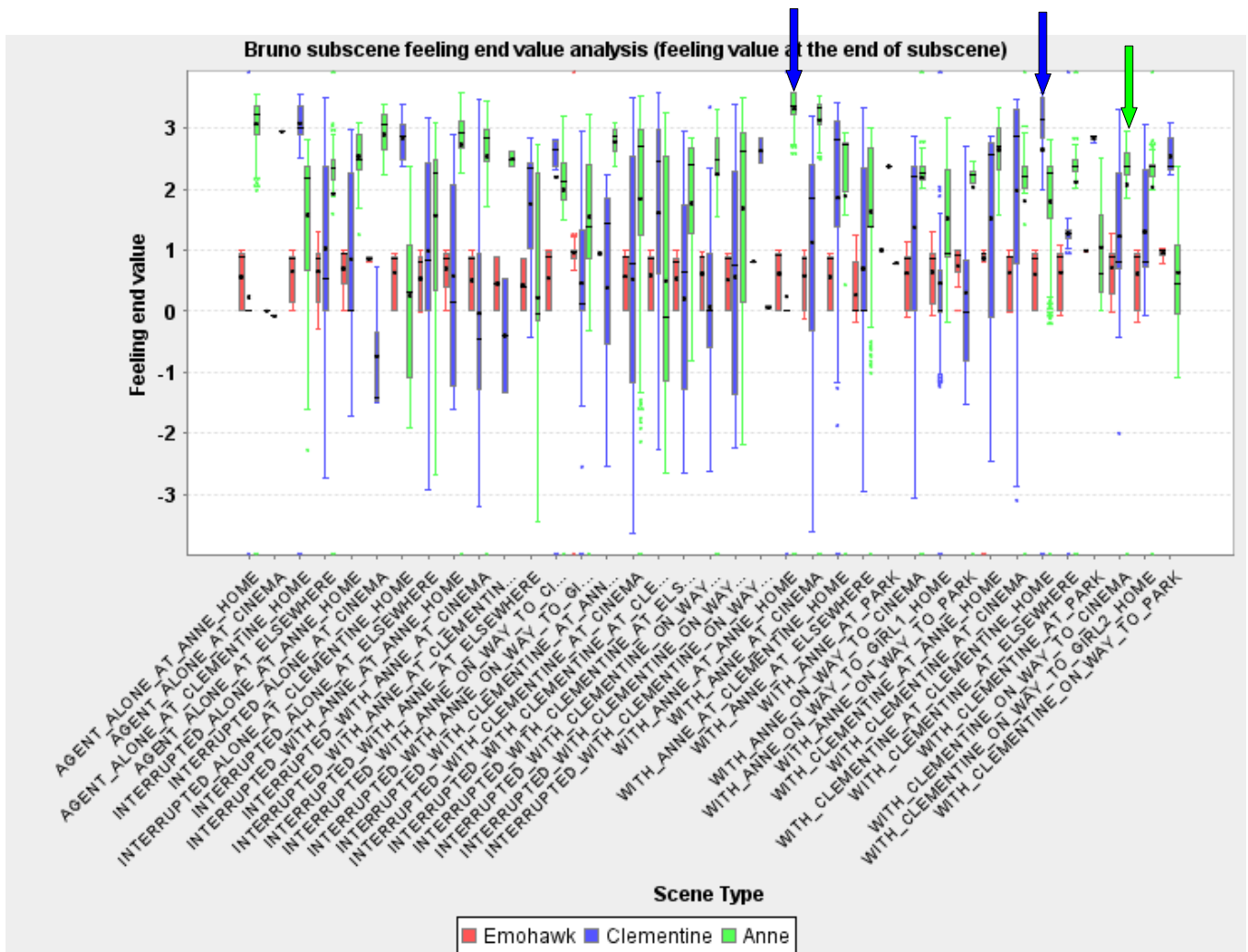
Graph 8. Bruno's sub-scene end feeling values. Graph is showing end feeling values for all the sub-scenes Bruno had experienced in the experiments.

Bruno's Graph 8 is containing a large amount of results. The values in this graph are more evenly distributed. This is due to the reason Bruno can experience both positive and negative

67

feelings with girls in the experiments. The resulting feeling values toward girls at certain times of the scenario are more emergent.

Marked by two red arrows we see strong positive trends toward girls in "at home" sub-scenes. Green arrow is marking other emergent trend of our scenario. On way to cinema with Clementine, Bruno has high feelings toward Anne as well. This is due to the fact that Bruno is going for Clementine right after he finishes his interaction with Anne and Anne is waiting at home for a while. The girls have no chance to meet at this time, so no conflicts take place and feeling value remains high.
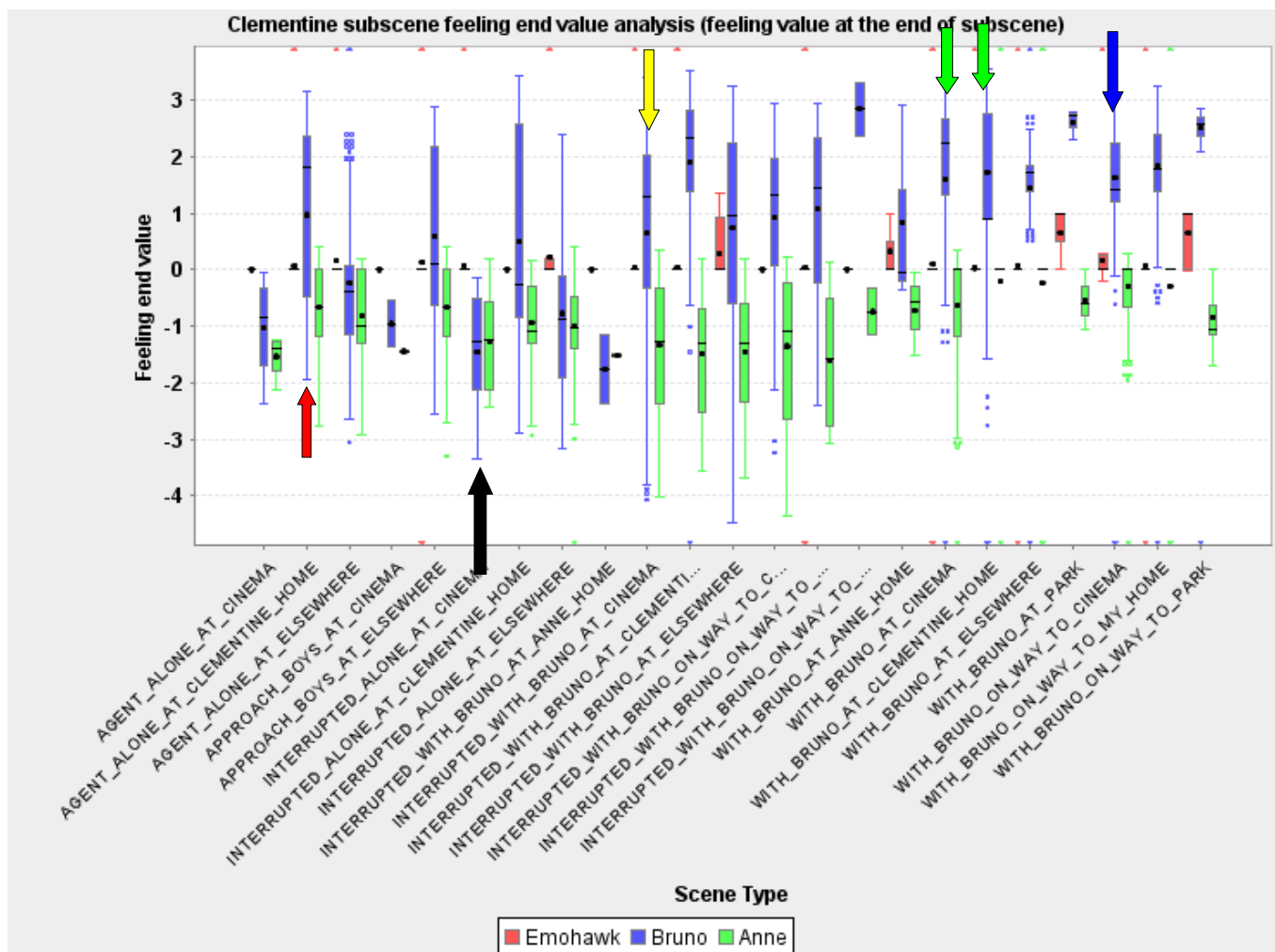
Graph 9. Clementine's sub-scene end feeling values. Graph is showing end feeling values for all the sub-scenes Clementine had experienced in the experiments.

Observing Clementine's Graph 9 we notice two green arrows showing positive feeling trends at situations where Clementine is alone with Bruno either at cinema or at home. Yellow arrow shows interesting trend – when Clementine was interrupted with Bruno at cinema, the conflict ended in more cases with Clementine having still positive feeling attitude toward Bruno. Red arrow shows that alone at her home, Clementine was usually positive towards Bruno. The reason for this is high probability of previous interaction with Bruno taking place at Clementine's home. These interactions usually end with positive feeling shifts. However a conflict may occur at Clementine's home as well (featuring Anne and lowering Clementine's feeling to Bruno).

The black arrow provides us with explication of the trend marked by green arrow on Graph 6. Actually the first reason (a) was true. After the conflict at cinema, feeling toward Bruno was shifting from higher negative values toward lower ones (the negative anger emotion was decaying – hence increasing the feeling value).

Summary. We have seen six graphs providing us with large number of values. We observed several trends in the experiments. "Interrupted" sub-scenes tend to produce more of negative feeling shifts, while sub-scenes "at home" quite the opposite. Generally speaking sub-scenes featuring three characters have bigger conflict potential than sub-scenes featuring two characters. Some of the less clear results on the graphs were due to small sub-scene occurrence and low data count. Also the emergent outcomes of agent to agent interactions had their effect especially on Bruno's graphs. For example considering Bruno interrupted sub-scenes, there is actually not clear, which of the girls he will dislike at the end of sub-scene and graphs do reflect that. Overall we have not observed any big abnormalities in these data.

Some of the box plots values were scattered across the graph reaching both positive and negative values. This is understandable considering that sub-scenes sequencing may differ between experiments as well as conflict occurrences. Conflict outcomes are partially emergent. This caused that sub-scenes featuring conflicts or sub-scenes occurring after conflicts usually do not show strong feeling trends (the values are scattered).

7.3.6   The feeling development in the experiment

We have performed a simple feeling development analysis in the experiments. First we will provide a brief summary of conflict occurrences for all the characters, after that we will present one feature of feeling development occurring in some experiments.

Conflict occurrences. We defined that character experienced a conflict, if his/her feeling value decreased to value -1 or below toward at least one other character. By committing a simple analysis, we have found out that a conflict occurred in the scenario:

– for Bruno in 242 cases (68%)

– for Anne in 269 cases (75%)

– for Clementine in 292 cases (82%)

Girls tend to experience higher number of conflicts due to the fact they generally dislike positive attitude of one of them toward Bruno. Also Bruno tends to experience higher positive feeling values toward girls making it harder for feeling to decrease to a conflict value (which we have tracked and which was selected to be minus one).

Usually our scenario produced one major conflict in the middle of scenario duration.

Feeling recovery problem. The feeling recovery problem is a situation where after a negative feeling shift, later in the experiment the feeling value recovers to positive values above one. This occurred in 10% (33) of all the experiments. Below we will plot three graphs showing this recovery problem for Bruno, Anne and Clementine.
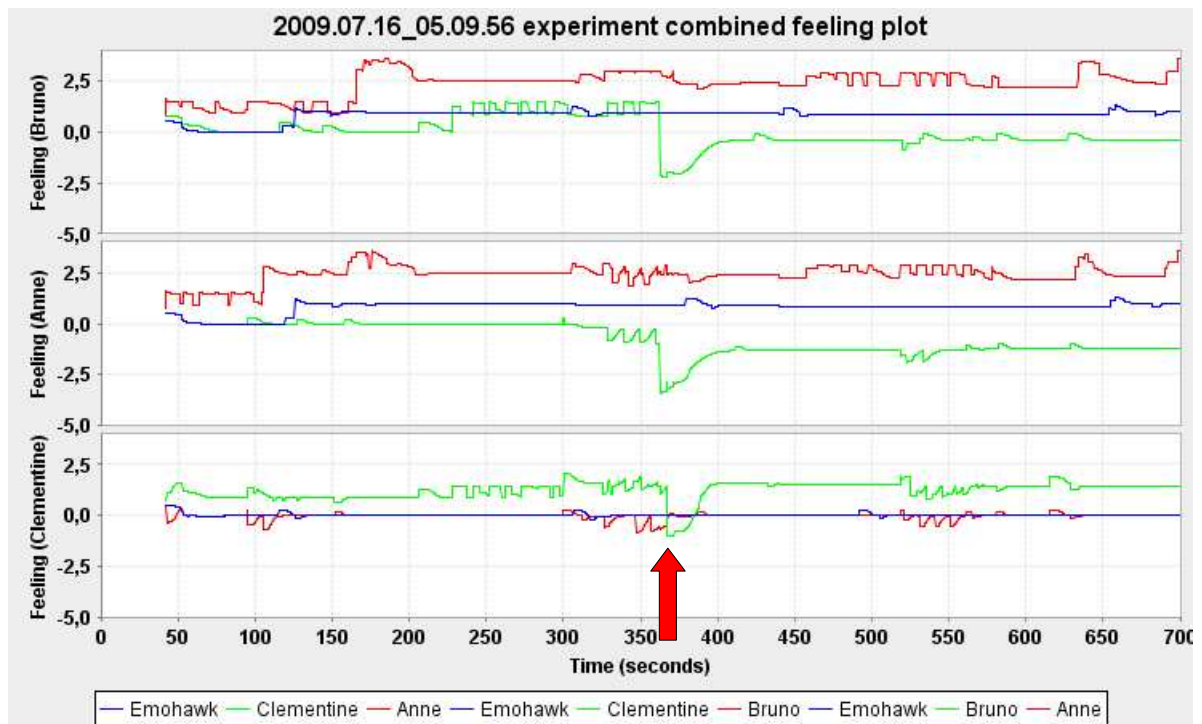
Graph 10. Clementine's feeling recovery problem. On the graph we see feeling plots of all three characters for experiment 2009.07.16_05.09.56.

The critical part on Graph 10 starts when Clementine and Bruno were interrupted by Anne. At that time, conflict occurred, resulting in Clementine slapping Anne. Bruno did not like this, so he performed negative action leave to Clementine (feeling shift marked by a red arrow). Feeling value of Clementine to Bruno decreased, however after some time it returned back to its initial value. This is due to the fact, action leave produces just temporary negative feeling shift. We consider this as a problem – the solution is that each time a negative action occurs at least some part of negative reaction should be permanent.
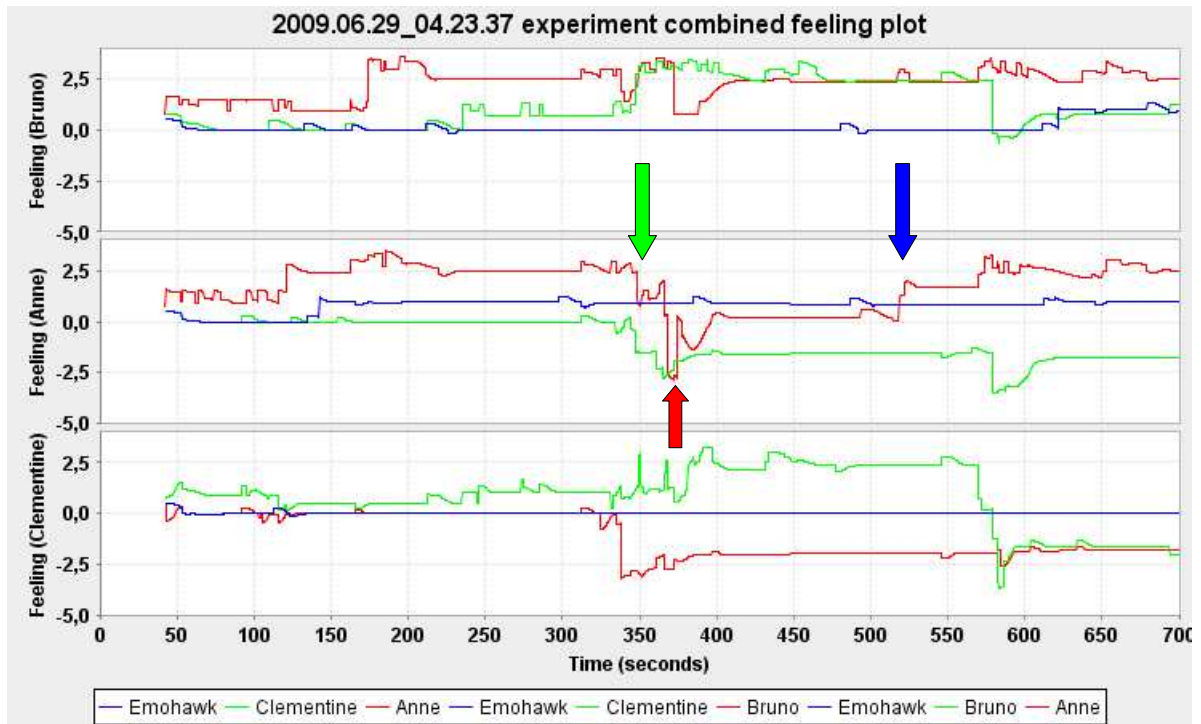
Graph 11. Anne's feeling recovery problem. On the graph we see feeling plots of all three characters for experiment 2009.06.29_04.23.37.

The start of problematic part on Graph 11 is marked by green arrow. It starts with Bruno being with Clementine, when Anne approached them. Anne witnessed proposals between Bruno and Clementine and even one kiss. Furthermore before that Anne accepted kiss proposal from Bruno, but the resulting kiss was appraised negatively, due to the feeling shift caused by other recent negative events. However few positive actions occurred between Anne and Bruno at that time as well. Nevertheless the feeling shifts here are quite quick and radical, going down and up (marked by a red arrow). This is undesired and points out we should revise our feeling computation algorithm. At the time marked by blue arrow, Bruno met Anne again and positive actions he performed resulted in positive feeling shift.
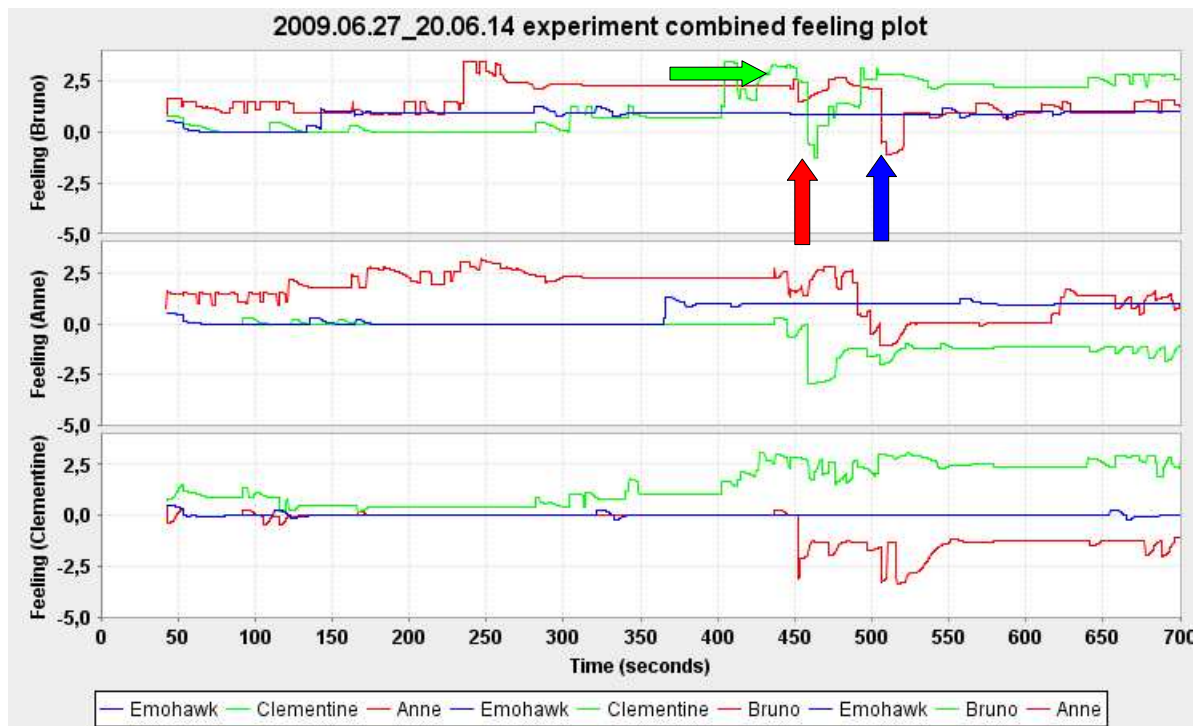
Graph 12. Bruno's feeling recovery problem. On the graph we see feeling plots of all three characters for experiment 2009.06.27_20.06.14.

The critical conflict started at time marked by green arrow on Graph 12. Bruno was at Clementine's home, when Anne arrived and interrupted Bruno and Clementine. Girls started to perform negative actions to each other causing feeling shifts at time marked by a red arrow. Bruno generally dislikes negative actions toward the ones he likes, so his feeling attitude to both girls decreased a bit (red arrow marking). Feeling to Clementine suffered more due to the fact Clementine used more negative action toward Anne (she slapped her). Clementine feelings toward Bruno remained high all the time, so she performed positive actions toward him. This caused positive feeling shift again. Anne than witnessed sex action between those two. This caused her to slap Bruno and relations were severed again (marked by blue arrow).

Now considering the problems here a) all of this was caused by allowing the girl to go to other girl's home, b) it is a question if we want to allow rapid down and up feeling shifts as shown on the graph around red arrow.

Summary. In some experiments we have witnessed that after a conflict, characters were able to resume the interaction later on in the scenario and achieve positive feeling values again. Besides, we have witnessed that our feeling development tend to feature up and down shifts sometimes of big magnitude. Actually the big up shifts are the bigger problem here.

Considering the believability it is not a problem if our agents are able to like each other even after some conflict occurs. It did not occur that much in the experiments and we could even want this to happen. However the way in which this was achieved featured sometimes not very natural feeling behavior. This was caused by a) by our scenario setting that allowed weird situation to occur – girl visiting other girl's home, b) event appraisal setting for some events and c) feeling computation algorithm behavior.

72

7.3.7 The feeling development after emotion events

Here we will present three graphs showing results of the feeling development after some emotion event took place for all of the characters Anne, Bruno and Clementine.
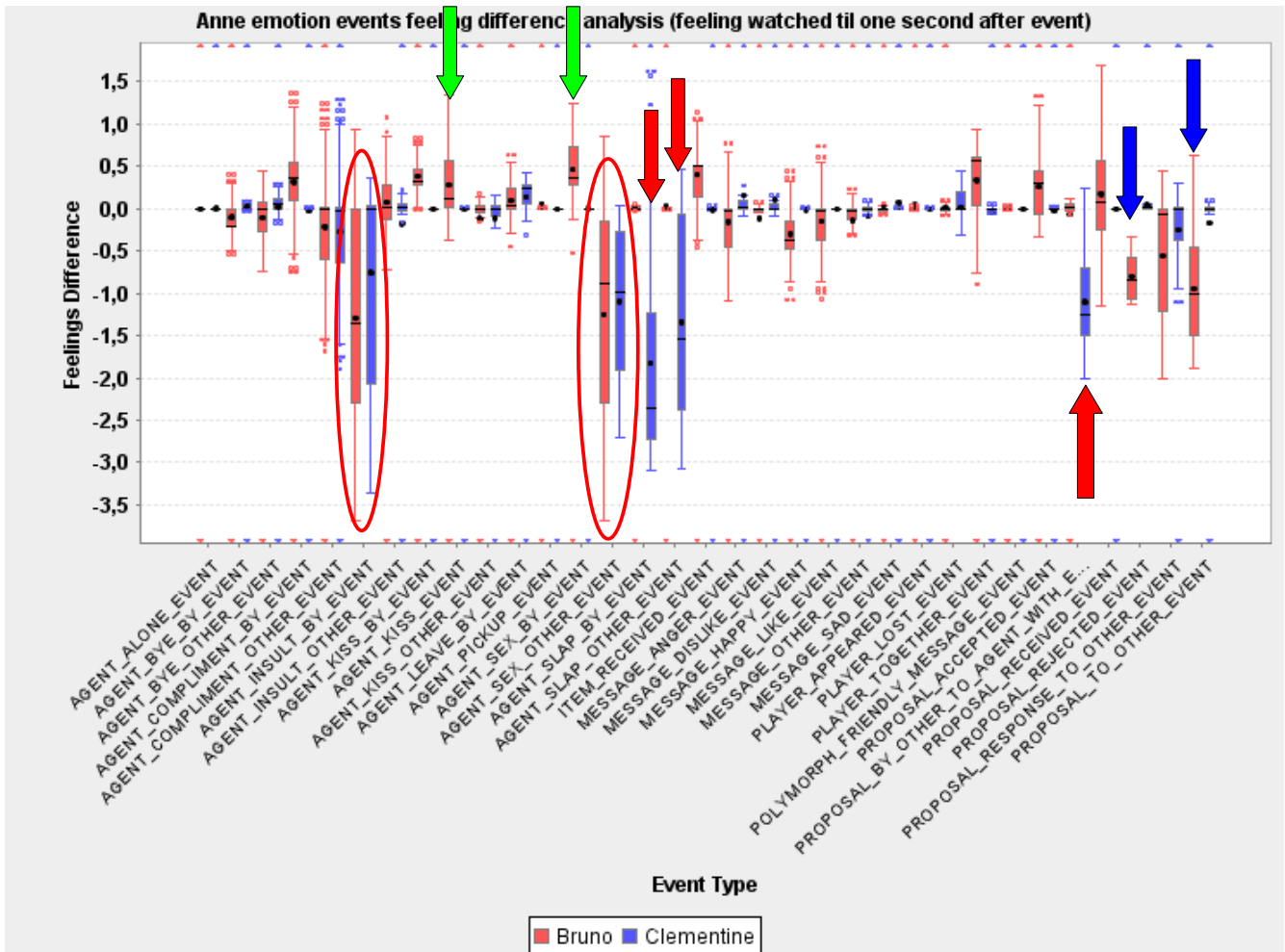
Graph 13. Anne's emotion events feeling difference plot. Graph is showing Anne's feeling shifts for emotion events. Feeling monitored from time of the event until one second after the event.

On Anne's Graph 13 we can observe shown by red ellipses that considering the negative actions that could be performed by both Bruno and Clementine the results between Bruno and Clementine are quite even. Red arrows are showing outcomes of negative emotion events that can be caused only by Clementine in the scenario. The blue arrows are showing outcomes of negative events that can be caused just by Bruno currently in the scenario. Green arrows are showing positive feeling shifts for actions kiss and sex.

The noise in the results is caused by the facts that a) there may be other appraised events in one second time we use to determine the event outcome and b) the feeling value may be in some extreme at the time of our event taking place biasing the feeling difference measure. However the Anne's feeling graph shows acceptable trends both for positive and negative events outcomes.
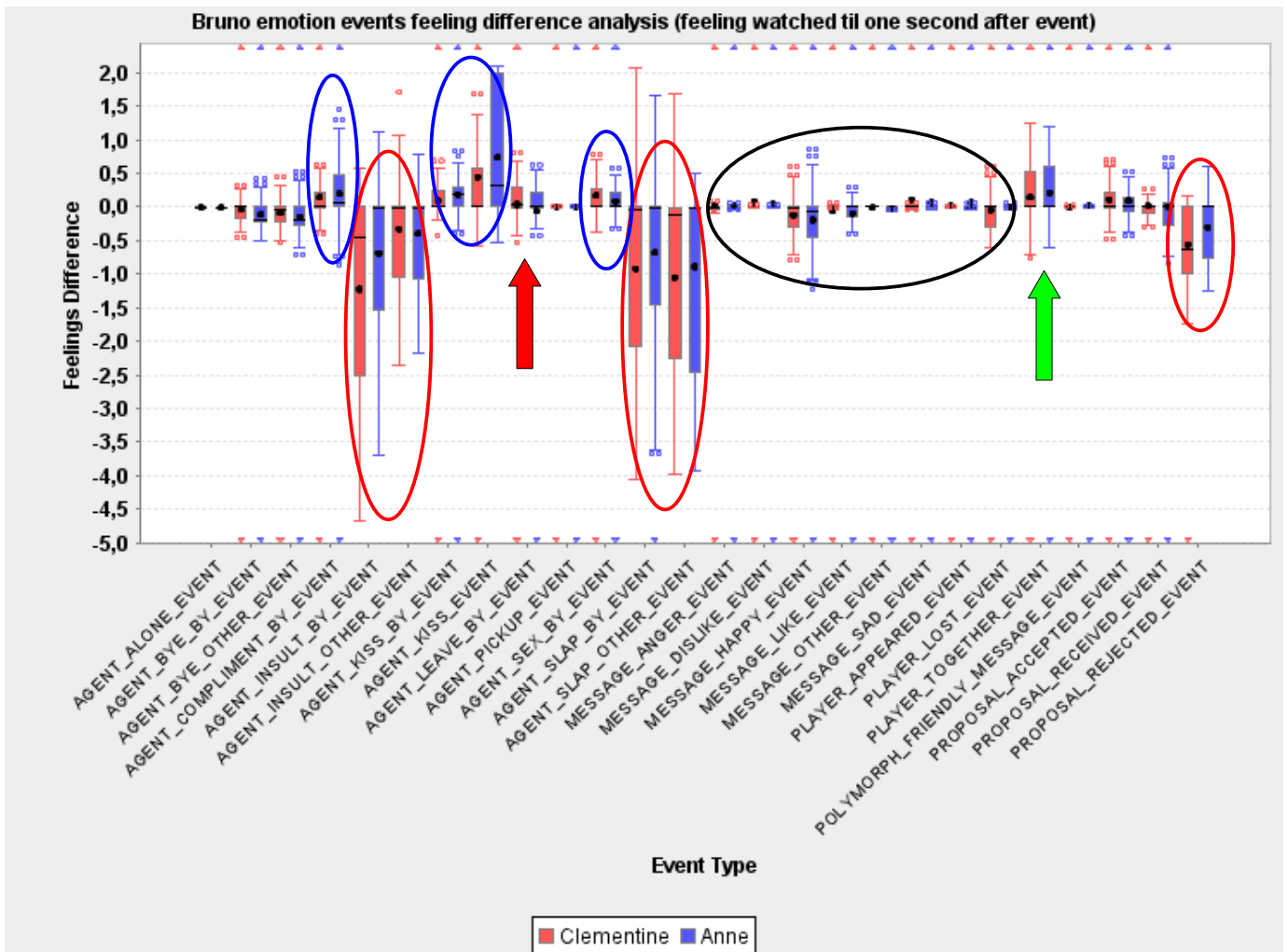
Graph 14. Bruno's emotion events feeling difference plot. Graph is showing Bruno's feeling shifts for emotion events. Feeling monitored from time of the event until one second after the event.

On Bruno's Graph 14 we can see the negative events marked by red ellipses produce usually negative feeling shifts, which is desired. The positive events marked by blue ellipses produce positive feeling shifts. Surprisingly "sex by event" does not produce as high feeling shifts as we would expect. This is due to the reason, that feeling is already high (above two), when the sex action can take place. Hence there is not much space for further increase.

By black ellipse we have marked message events that do not produce big feeling changes (as intended). Marked by green arrow we can notice "player together event" that seems to produce more positive than negative outcomes. Both of the outcomes are possible for this event – it depends on feeling value toward the character we are together with. Red arrow marks leave by action, that produces more feeling increases than decreases – this is due to positive appraisal of leave action, when the our feeling value to other agent is negative. This actually could be a problem. The negative actions should produce only negative outcomes.
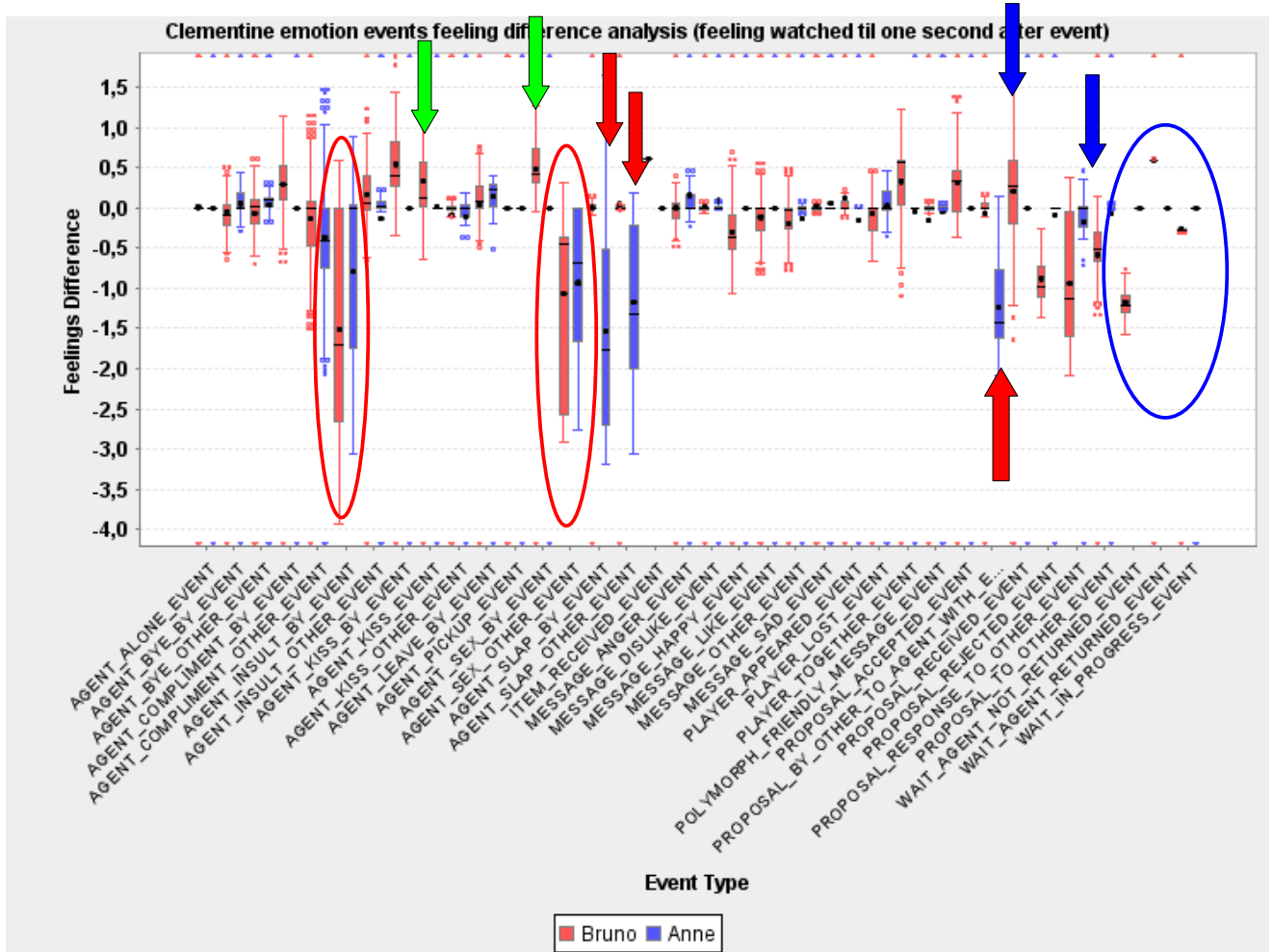
Graph 15. Clementine's emotion events feeling difference plot. Graph is showing Clementine's feeling shifts for emotion events. Feeling monitored from time of the event until one second after the event.

On Clementine Graph 15 red ellipses show negative events that can be performed by both Bruno and Anne. Red arrows are marking negative events that can be performed only by Anne, blue arrows are showing event that can be performed only by Bruno. We can see that the trends here resemble the trends on Anne's Graph 13.

Furthermore blue ellipse is showing results of Clementine's waiting events. Due to the fact that no agent interaction usually takes place before these events take place for Clementine, the results are noticeably tight here. Negative event "agent not returned" showing a negative shift for Bruno, while positive action "agent returned" a positive shift.

Summary. Graphs here show rather desirable outcomes – negative events tend to produce negative feeling shifts and positive events vice versa. Some problems however were noticed with action leave that did not produce always negative feeling shift (this is a problem of appraisal). Also the positive action sex did not produce higher feeling increases. The reason is that emotions produced by this action (liking and love) are already produced by actions usually preceding action sex (e.g. action kiss). Hence there is a smaller space for feeling increase. This is a point for discussion if this is desirable or not.

7.3.8 The action occurrences

In this sub section we will present results considering the counts of actions in the experiments. Firstly we will present global statistics – count of actions in all of the experiments for all the characters and secondly we will present results considering the action key points in the experiments.

In the tables below we use shortcuts for the actions. Table 13 presents shortcuts for all displayed actions. We will continue to use these shortcuts later on as well.

B	Bye action
CO	Compliment action
CU	Cuddle action
I	Insult action
KS	Kiss action

L	Leave action
P	Proposal action
SL	Slap action
SX	Sex action
T	Talk action (conversation)

Table 13. Action shortcuts. This table shows shortcuts we use for scenario actions.

Note that kick action is not included in the shortcut lists – it is due to the fact that the kick action did not occur in the experiments – the emohawk was programmed to be friendly, so the characters had no desire to select negative action kick when interacting with him.

Global action statistics. Following three tables show global action statistics for all three characters.

	Emohawk
B	69
CO	0
CU	108
I	0
KS	0
L	0
P	0
SL	0
SX	0
T	321

	Clementine
B	698
CO	2642
CU	0
I	159
KS	468
L	103
P	2245
SL	0
SX	319
T	7788

	Anne
B	994
CO	4729
CU	0
I	116
KS	674
L	36
P	2464
SL	0
SX	390
T	11052

Table 14. Bruno's global actions statistics. Table shows Bruno's action counts toward all three possible targets in the scenario. Reason why Bruno is not performing action slap toward girls due to the design decision.

Table 14 shows that the feeling end value results with Anne being a favorite girl of Bruno are supported also by action counts. The table portrays overall higher action counts toward Anne. The only two actions that feature higher count for Clementine are the negative actions insult

and leave.

	Emohawk
B	6
CO	0
CU	15
I	0
KS	0
L	0
P	0
SL	0
SX	0
T	60

	Bruno
B	461
CO	3017
CU	0
I	108
KS	262
L	122
P	408
SL	151
SX	365
T	9339

	Anne
B	280
CO	0
CU	0
I	71
KS	0
L	158
P	0
SL	148
SX	0
T	601

Table 15. Clementine's global actions statistics. Table shows Clementine's action counts toward all three possible targets in the scenario. Girls do not perform positive actions toward each other as can be seen in the table on the right. The asymmetry connected with action sex between Clementine and Bruno is caused by the message loss problem (Bruno overheard a portion of accepted sex proposals).

In the Clementine's global action statistics Table 15 we can observe that girls do not perform positive actions toward each other as well as proposals. Clementine performed slightly more negative actions toward Bruno than to Anne with the exception of action leave. This is caused by the fact, that the girl to girl interaction takes place exclusively in interrupted state. This state is programmed to end automatically after some time, which results in saying goodbye – if the action bye or action leave will be selected – that depends on the actual feeling value toward a character. As the girls feature more negative affect toward each other, they tend to use action leave for saying goodbye to each other.

	Emohawk
B	83
CO	0
CU	168
I	0
KS	0
L	0
P	0
SL	0
SX	0

	Bruno
B	767
CO	5256
CU	0
I	64
KS	402
L	84
P	504
SL	105
SX	451

	Clementine
B	281
CO	0
CU	0
I	85
KS	0
L	148
P	0
SL	146
SX	0

T	600	**T**	12730	**T**	569

Table 16. Anne's global actions statistics. Table shows Anne's action counts toward all three possible targets in the scenario. The asymmetry connected with action sex between Anne and Bruno is caused by the message loss problem (Bruno overheard a portion of accepted sex proposals).

Now when we observe Anne's global actions statistics in Table 16 it is a good time to make a comparison between the two girls. Anne being Bruno's favorite hypothesis is confirmed here as well by higher positive action count by Anne to Bruno than by Clementine to Bruno in Table 15. Girl to girl interaction seems to end fairly symmetric. It seems that girls tend to react to negative actions negatively. This seems appropriate.

Considering the emohawk the highest action count toward him is achieved by Anne, second is Bruno and third Clementine. Anne and Bruno high action count toward emohawk is given by the fact, that when Bruno walks with Anne toward Anne home at the beginning of the scene, the path leads right next to the emohawk. So the emohawk has higher probability to interrupt Bruno and Anne than Clementine, who rarely walks around him.

We would like to note here, that due to the unreliable communication Bruno actually missed 9% of accepted love proposals by Anne and 11% of accepted love proposals by Clementine causing the asymmetry in action sex counts.

Key point – intimate actions in the experiments. Here we will inspect the experiments counts that ended with intimate action (action sex) taking place between Bruno and the girls. Achieving intimate relation with a girl is one of the key points in our scenario in our prototype implementation (intimate action has the highest feeling requirements).

	Intimate with A	**Intimate with C**	**NOT intimate with A**	**NOT intimate with C**
Intimate with A	296 (83%)	191 (54%)	-	105 (29%)
Intimate with C	191 (54%)	232 (65%)	41 (11%)	-
NOT intimate with A	-	41 (11%)	61 (17%)	20 (6%)
NOT intimate with C	105 (29%)	-	20 (6%)	125 (35%)

Table 17. Intimate action occurrence (action sex) in the experiments. Table shows results for Bruno achieving intimate relations with the girls. Intimate with A means Bruno achieved intimate relation with Anne (action sex occurred), intimate with C means Bruno achieved intimate relation with Clementine. Keyword NOT represents negation. We would like to note that 6% of experiments did not feature any intimate actions.

Summarizing Table 17 – in more than a half of all the experiments Bruno managed to establish love relation with both of the girls. This contrasts with the fact, that considering the end feeling relations, just in 19% of all the experiments Bruno was in love with both girls and both girls were in love with Bruno. The reason is that after successfully achieving love relation with a girl a conflict may occur including the other girl. The feeling relations may be shattered by this. Since our scenario was written in a way that attracts conflict occurrences,

this is nothing unexpected.

7.3.9   The actions sequencing – action pairs

Here we will analyze action pairs in the scenario. These pairs are defined as follows. Two actions performed by character A to character B in a row are considered a pair if the delay between them does not exceed 15 seconds. The other actions performed in between to other characters do not break this rule. Hence we construct pairs of actions toward target characters.

We allowed these action types to be in action pairs: bye, compliment, cuddle, insult, kiss, leave, slap and sex. Moreover we performed an analysis where we added proposals to this pool (all proposals were treated as "action" proposal). We will present two types of results – results without action proposal and with action proposal. The reason to consider proposals as action is that proposals always precede certain actions (e.g. characters will always ask – make proposal – before action kiss). We were interested if this will be shown by the results.

The tables below are read from left to right as follows. We always start action pair with the first action in a row, each column defines the second action of a pair. Values in the tables are the actual occurrences of these action pairs in all of the experiments.

	B	CO	CU	I	KS	L	SL	SX
B	**14**	**5**	0	0	0	**1**	0	0
CO	157	1495	0	16	241	17	29	297
CU	4	0	8	0	0	0	0	0
I	84	0	0	10	0	32	18	0
KS	1	154	0	0	0	2	0	2
L	**2**	0	0	0	0	0	**1**	0
SL	118	**1**	0	13	0	66	56	0
SX	146	158	0	0	0	0	0	0

Table 18. Clementine's action pairs excluding action proposal. Suspicious results are marked by bold, underlined font.

	B	**CO**	**CU**	**I**	**KS**	**L**	**P**	**SL**	**SX**
B	<u>**14**</u>	<u>**5**</u>	0	0	0	<u>**1**</u>	0	0	0
CO	157	1408	0	13	37	15	342	24	297
CU	4	0	8	0	0	0	0	0	0
I	84	0	0	10	0	32	0	18	0
KS	1	154	0	0	0	2	0	0	2
L	<u>**2**</u>	0	0	0	0	0	0	<u>**1**</u>	0
P	0	119	0	3	221	5	5	6	0
SL	118	<u>**1**</u>	0	13	0	66	0	56	0
SX	146	158	0	0	0	0	0	0	0

Table 19. Clementine's action pairs including action proposal. Suspicious results are marked by bold, underlined font.

On tables 18 and 19 we can see that the including action proposal did not affect Table 19 results much with the exception of proposal, kiss action pair. This is obvious due to the fact that our characters in the scenario always ask before they kiss someone.

The suspicious results here are definitely all negative actions followed by positive action (marked by bold, underlined font) – especially the pair slap, compliment. Also every pair of action starting with action bye is suspicious (marked by bold, underlined font) – why we would say bye to someone twice in a row? There are two reasons for action pairs starting with bye and leave.

a) Message loss effect – sometimes the other agent simply overhears our action bye and will continue to interact with us. This results in performing second action bye toward him.

b) Double interruption effect – when three agents are interacting, there may a situation occur that two characters are saying bye to each other, but they are still being interrupted by third character and continue interaction with him. This can potentially lead to a situation, where they will interrupt the agent, they said bye to before, again. We will illustrate this effect on an example in suspicious experiment analysis section (7.4).

We have not treated action pairs with positive and negative actions in a row as suspicious due to the reason that rapid feeling decreases are "normal" in our scenario due to the number of conflicts that occur in it. These rapid feeling decreases are the causes of action pairs with positive and then negative actions.

	B	CO	CU	I	KS	L	SL	SX
B	**276**	**51**	0	**1**	**3**	**1**	0	0
CO	292	4197	0	56	758	35	0	191
CU	24	57	0	0	0	0	0	0
I	154	**5**	0	33	0	26	0	0
KS	2	534	0	0	0	0	0	409
L	**9**	0	0	**1**	0	**7**	0	0
SL	0	0	0	0	0	0	0	0
SX	684	10	0	0	0	0	0	0

Table 20. Bruno's action pairs excluding action proposal. Suspicious results are marked by bold, underlined font.

	B	CO	CU	I	KS	L	P	SL	SX
B	**268**	**51**	0	**1**	0	**1**	**13**	0	0
CO	291	3691	0	48	378	8	2327	0	47
CU	24	0	57	0	0	0	0	0	0
I	153	**5**	0	33	0	25	2	0	0
KS	2	335	0	0	0	0	784	0	0
L	**9**	0	0	**1**	0	**7**	0	0	0
P	18	1364	0	30	763	45	30	0	662
SL	0	0	0	0	0	0	0	0	0
SX	684	10	0	0	0	0	0	0	0

Table 21. Bruno's action pairs including action proposal. Suspicious results are marked by bold, underlined font.

On Bruno's tables 20 and 21 we can clearly see what impact had the inclusion of action proposal in the analysis. Since proposal precedes each action sex for Bruno, the pair kiss, sex completely vanished from the results in Table 21! Also we can see high number of proposal, kiss action pair – again the reason is Bruno always asks before he makes the kiss. Other numbers in the table are not that affected by inclusion of action proposal. We can observe on the tables that Bruno never performs action slap.

High number of sex, bye pair illustrates the fact that Bruno ends interaction with a girl after sex action between him and the girl occurs.

	B	CO	CU	I	KS	L	SL	SX
B	**_4_**	**_6_**	0	0	0	0	**_3_**	0
CO	419	3158	0	4	375	3	8	386
CU	43	0	67	0	0	0	0	0
I	77	0	0	11	0	30	10	0
KS	14	238	0	0	0	1	0	1
L	**_1_**	0	0	0	0	0	0	0
SL	81	**_1_**	0	27	0	62	44	0
SX	156	279	0	0	0	0	0	1

Table 22. Anne's action pairs excluding action proposal. Suspicious results are marked by bold, underlined font.

	B	CO	CU	I	KS	L	P	SL	SX
B	**_4_**	**_6_**	0	0	0	0	0	**_3_**	0
CO	408	3054	0	2	67	3	448	8	386
CU	43	0	67	0	0	0	0	0	0
I	77	0	0	11	0	30	0	10	0
KS	14	238	0	0	0	1	0	0	1
L	**_1_**	0	0	0	0	0	0	0	0
P	12	133	0	4	329	0	0	1	0
SL	81	**_1_**	0	27	0	62	1	44	0
SX	156	279	0	0	0	0	0	0	1

Table 23. Anne's action pairs including action proposal. Suspicious results are marked by bold, underlined font.

Observing tables 22 and 23 we see some nice trends in our scenario. The action pair slap, leave or insult, bye are nice examples how Clementine is solving conflict situations – first she performs negative action and then she leaves (similar trend can be seen also in Anne's tables 18, 19).

Summary. We have seen action pair statistics in this sub section. We have found that action pairs are affected by message loss effect and double interruption effect. Overall our scenario did not feature high number of suspicious action pairs – those are especially negative actions followed by positive ones. The opposite phenomen is normal due to the conflict occurrences. Some of the experiments containing suspicious action pairs will be analyzed in detail in suspicious scenario analysis section.

7.3.10 The sub-scene sequencing

Due to the high number of result, just brief summary will be presented here. Firstly we will summarize sub-scene series analysis. And secondly we will regard sub-scene pairs analysis.

Note that each character in the scenario has his own set of sub-scenes for one experiment.

Sub-scenes series analysis. The goal of this analysis was to pinpoint whether there are some usual sequences of sub-scene series generated by our scenario. Considering this analysis we encountered following problems:

a) Due to the nature of our sub-scene generation algorithm sometimes we get in sub-scene series a sequence of two same sub-scenes. This is a result of the fact that as our agents traverse through environment, sometimes they pass our recognized places such as cinema. This causes for example that sub-scene "ALONE_AT_ELSEWHERE" will be briefly interrupted by sub-scene "ALONE_AT_CINEMA". Since the character will go over this place quickly the needed five seconds threshold will not be reached for sub-scene to be generated. Hence a sequence of two sub-scenes "ALONE_AT_ELSEWHERE" in a row will be generated. This can be solved for example by post-processing of sub-scene strings and joining these sub-scene pairs.

b) Our methodology of comparison did not allow any tolerance in matching of two sub-scene series. However some tolerance could help to regard two fairly similar sub-scene series as one.

Despite the down sides of our sub-scene series analysis, we have observed some sub-scene series occurred more than once. E.g. for Anne following sub-scene series occurred in 2% of all the experiments (8 occurrences). It also illustrates the problem of same sub-scene succeeding each other:

1) WITH_BRUNO_AT_ELSEWHERE; 2) WITH_BRUNO_ON_WAY_TO_MY_HOME; 3) WITH_BRUNO_AT_ANNE_HOME; 4) AGENT_ALONE_AT_ANNE_HOME; 5) APPROACH_BOYS_AT_ELSEWHERE; 6) APPROACH_BOYS_AT_ELSEWHERE; 7) WITH_BRUNO_ON_WAY_TO_CINEMA; 8) WITH_BRUNO_AT_CINEMA; 9) WITH_ BRUNO_ON_WAY_TO_MY_HOME; 10) AGENT_ALONE_AT_ANNE_HOME; 11) AGENT_ALONE_ AT_ANNE_HOME;

Sub-scene pairs analysis. Considering sub-scene pairs we would like to pinpoint high number of occurrences of simple story sub-scene pairs such as:

WITH_BRUNO_ON_WAY_TO_CINEMA; WITH_BRUNO_AT_CINEMA – 472

WITH_BRUNO_AT_CINEMA; WITH_BRUNO_ON_WAY_TO_MY_HOME – 403

Below are some other sub-scene pairs with high count of occurrence. Commenting in the order of appearance – a) an encounter with a girl in the environment, b) girl leaving her home after some time and c) Bruno taking girl home.

AGENT_ALONE_AT_ELSEWHERE; WITH_CLEMENTINE_AT_ELSEWHERE – 479

AGENT_ALONE_AT_CLEMENTINE_HOME; APPROACH_BOYS_AT_ELSEWHERE – 384

WITH_BRUNO_AT_ELSEWHERE; WITH_BRUNO_ON_WAY_TO_MY_HOME – 633

We have also performed an analysis of suspicious double sub-scenes in the experiments. These were usually rarely occurring sub-scene pairs that did not seem logical at the first glance. The reasons for these suspicious sub-scenes pairs were:

a) message loss – sometimes proposal responses or actions were not heard

b) fast interaction – rapid change of agent states did not allow for all the occurring sub-

scenes to be captured

c) agents got stuck somewhere

The suspicious sub-scenes pairs were found in ~ 17% of all the experiments. Below we will show and comment some of the suspicious sub-scenes pairs in the experiments.

WITH_BRUNO_ON_WAY_TO_MY_HOME; INTERRUPTED_ALONE_AT_ANNE_HOME; (Anne, experiment 2009.06.28_00.40.52)

Bruno walked Anne home, but Anne did not reach the home place, so she remained in previous sub-scene. Bruno ended the interaction at Anne's home by saying bye, but at that time Clementine arrived and interrupted Bruno and Anne. Since Bruno said bye to Anne recently, she did not select him as "agent with", but treated him as an interrupter as well.

WITH_ANNE_AT_ANNE_HOME; INTERRUPTED_ALONE_AT_ANNE_HOME; (Bruno, experiment 2009.06.28_08.04.28)

Bruno walked Anne at home and after some time, he said bye to Anne. Due to message loss Anne overheard this message and continued the interaction. Since Bruno said bye to Anne recently he did not select her as "agent with" at this point, but treated her as an interrupter. After some time Bruno said bye to Anne again. This time Anne heard it and Bruno departed.

AGENT_ALONE_AT_CINEMA; APPROACH_BOYS_AT_CINEMA; (Clementine, experiment 2009.06.29_09.38.48)

After conflict occurred at cinema. Two characters – Clementine and Bruno – were stuck together at the cinema. This prevented Clementine to actually go home as she was programmed to. After some time, approach boys state was triggered for Clementine. This had finally unstuck the agents.

WITH_BRUNO_ON_WAY_TO_CINEMA; AGENT_ALONE_AT_CLEMENTINE_HOME; (Clementine, experiment 2009.07.15_21.34.46)

On the way to cinema from Clementine's home, Bruno and Clementine met Anne. The interaction between agents was hectic – some of the sub-scenes did not exceed the five seconds threshold to be generated. It all ended by Clementine staying at home after the conflict with Bruno.

WITH_BRUNO_ON_WAY_TO_CINEMA; INTERRUPTED_WITH_BRUNO_ON_WAY_TO_ MY_HOME (Clementine, experiment 2009.07.16_09.15.04)

This sub-scene pair was triggered by the message loss. Bruno overheard accepting of the first cinema proposal. As another proposal, he selected proposal to go home, which was accepted by Clementine as well. On the way, they met Anne and this triggered the interrupted state.

7.3.11 The sub-scene durations

Here we will present a graph with overview of sub-scenes duration in the experiments.

84

Currently we do not have any demands onto durations of the sub-scenes in the scenario, so this result serves reviewing purpose. However we have observed a few trends even here. The occurrence of blank places was minimal across all the experiments.
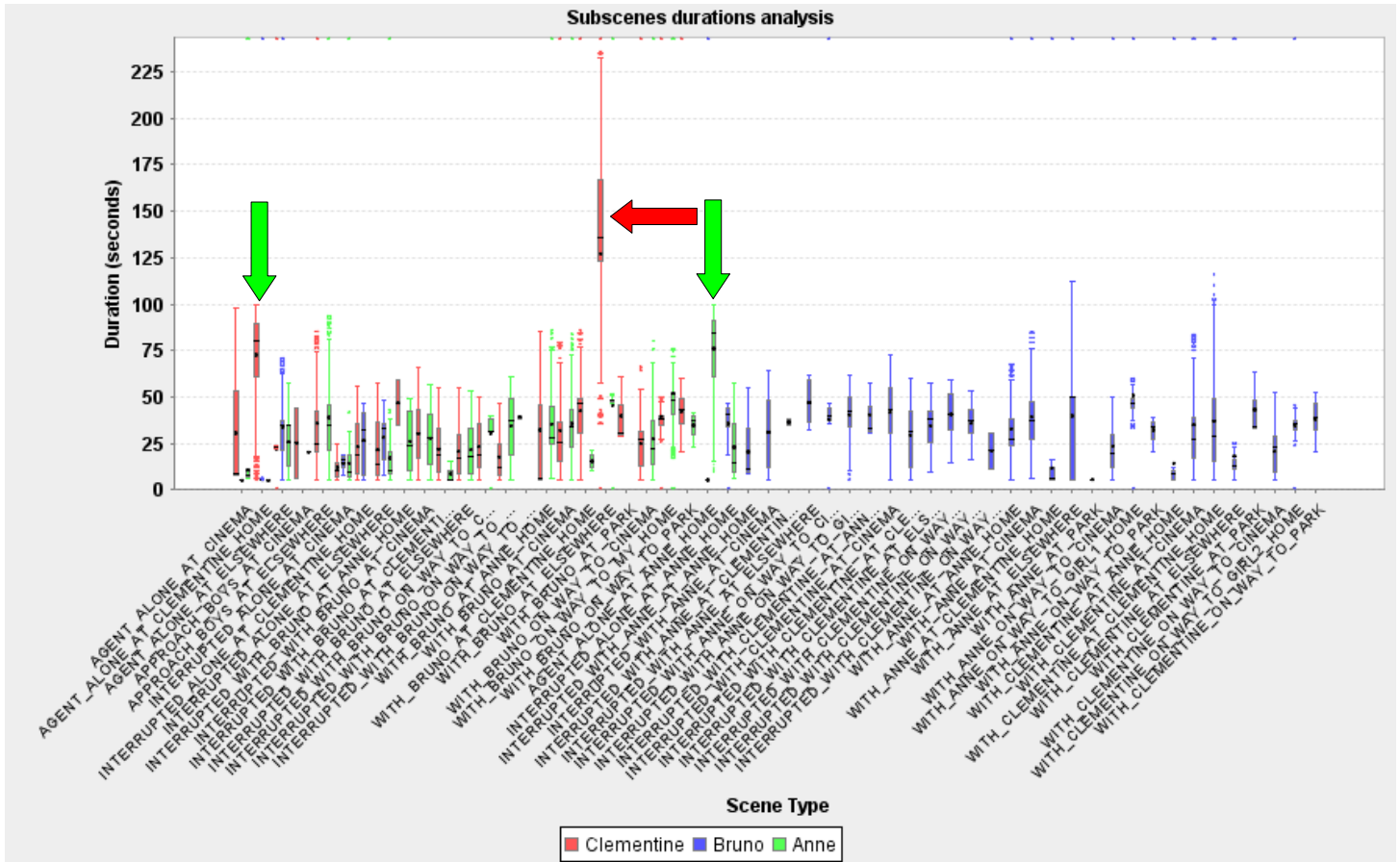
Graph 16. Sub-scenes durations. Graph is plotting box plots with durations of each sub-scene in the scenario.

Green arrows on Graph 16 show the increased durations of sub-scenes "Anne alone at her home" and "Clementine alone at her home". The girls are programmed to wait at home for a while, when they reach it. This is clearly visible here. The red arrow is showing very long duration for Clementine's sub-scene "with Bruno at elsewhere". This is actually due to the implementation of waiting for someone in the experiment. When Clementine is waiting for Bruno at the beginning of the experiment, Clementine has her "agent with" variable set to Bruno. So, even she is not with him at the moment, considering the sub-scenes it looks like as she would be with him. This is actually rather undesired effect that should be fixed in future work.

The durations of rest of the sub-scenes are quite even with mean values ranging from 25 to 50 seconds.

7.4   *Suspicious experiments analysis*

Here we will analyze selected experiments in a greater detail. Firstly, we illustrate the double interruption problem connected with the occurrence of action pair bye, slap. Secondly, we review two experiments featuring slap, compliment action pair. Thirdly, we look upon two

experiments featuring asymmetric relations between characters. Two last experiments will show rather unpleasant features of our project Emohawk prototype implementation discovered during the evaluation.

7.4.1   Double interruption problem and action pair bye, slap

The analysis of experiment 2009.06.27_15.37.46 shows how the double interruption problem leads to occurrence of suspicious action pairs such as bye, slap pair.
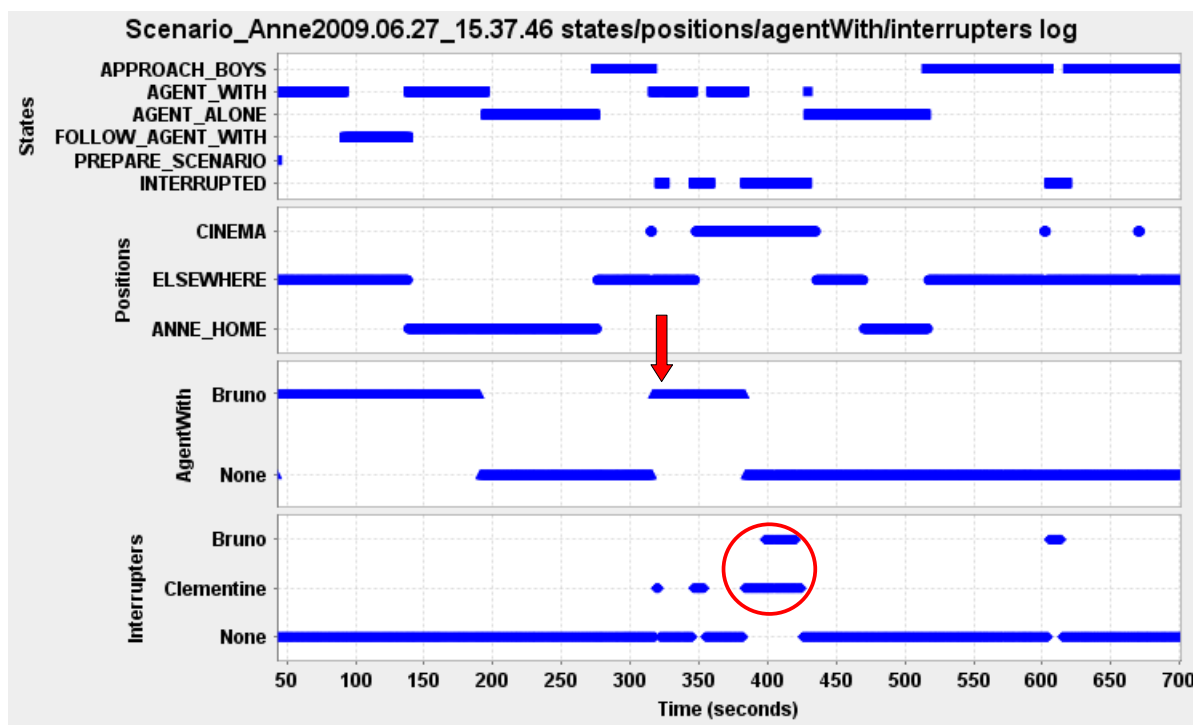
Graph 17. Anne's state diagram. The critical areas on the graph are marked by a red circle and a red arrow.

The critical part of the story starts at the point marked by a red arrow on Graph 17. Anne met Bruno with Clementine at cinema. Anne automatically starts to interact with Bruno immediately selecting him as her "agent with", because he is a boy that she likes. However, Bruno is here with Clementine, so he shifts to interrupted state. After some time Bruno decided to end the interaction with Anne and used action bye – this can be seen on the Anne's state graph – it is the time when blue line marked by a red arrow ends. At that time Anne is already interrupted by Clementine (marked by a red circle). Clementine started to interact with Anne, because Anne was interacting with Bruno and Bruno was Clementine's "agent with". Due to the same reason, after some time Bruno will resume interaction with Anne, because Anne is interacting with Bruno's "agent with" – Clementine (also marked by a red circle). At this time Anne feeling value toward Bruno is fairly severed, due to all the positive actions Bruno managed to exchange with Clementine. This leads her to selection of slap action completing the bye, slap action pair toward Bruno. As we observe this is nothing too much unbelievable.

On the feeling development Graph 18 below from the same experiment we can see the start of the interaction at cinema marked by green arrow – after that we can observe on Anne's

sub-graph negative feeling shift resulting in action slap – marked by a red arrow.
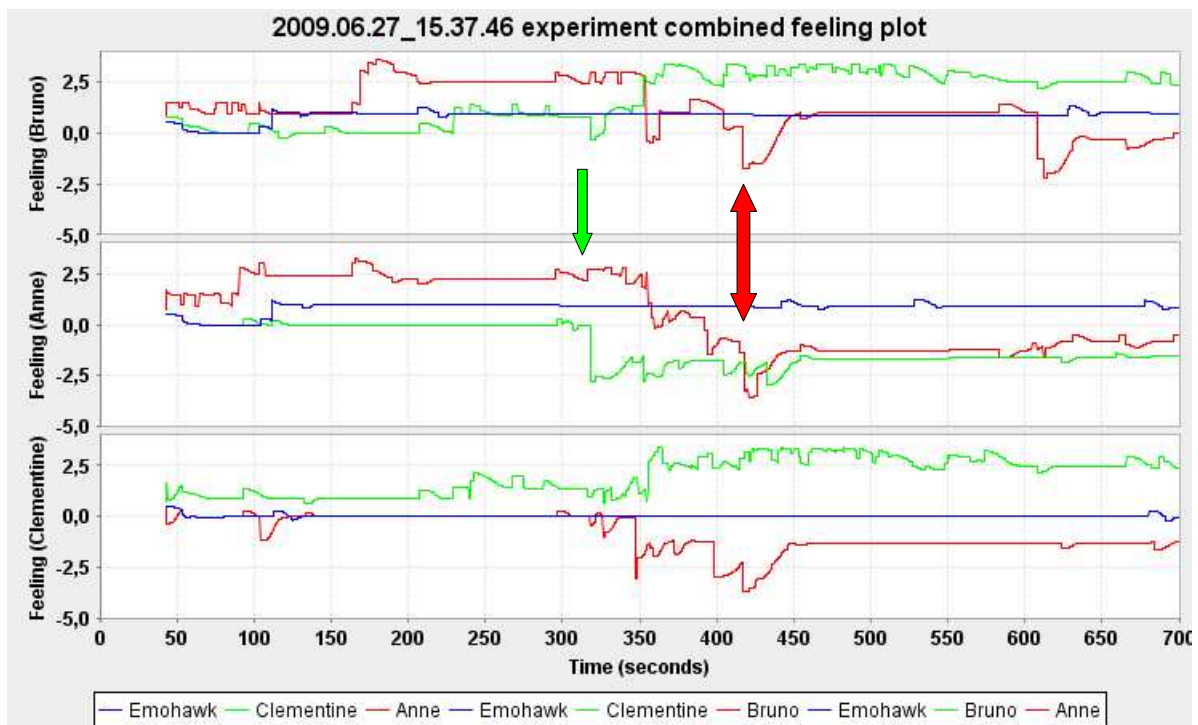
Graph 18. Combined feeling development plot. Combined feeling development plot from experiment 2009.06.27_15.37.46. Bruno feeling development on upper graph with Anne marked by red line. Anne feeling development on middle graph with Bruno marked by red line.

7.4.2   Action pair slap, compliment experiments

Here we will analyze two experiments featuring action pair slap, compliment.

Experiment 2009.06.29_03.13.35. On Graph 19 we can see the Clementine's feeling development in the experiment. She performed action slap toward Bruno at time marked by a red arrow. After that a positive event occurred increasing feeling enough for action compliment to be selected at time marked by green arrow. Note that for action slap feeling needs to be below -1. For action compliment the feeling value needs to be above 0.3.

Solution to this problem may be to re-balance feeling values triggering the actions or optimize the algorithm for the feeling computation.

Experiment 2009.07.14_07.34.11. Here Anne interrupts Bruno with Clementine at cinema. Anne feeling value decreases due to proposals made by Bruno to Clementine (marked on Graph 20 by a red arrow). The result of this is action slap toward Bruno. But right before the negative feeling shift Anne proposed a kiss toward Bruno. The response to kiss proposal and the action slap occur approximately at the same time, so kiss proposal is accepted by Bruno resulting in action kiss between Bruno and Anne increasing the feeling value for Anne again. This later results in positive action compliment issued toward Bruno.

This experiment illustrates feeling recovery problem mentioned in section 7.3.6 above. This problem occurred in ~ 10% of all the experiments. On this example experiment we have seen, that when the timing of actions and proposals is correct, there may occur rather weird interactions between agents, which are not believable. These issues may be solved by a) redesigning the feeling value computation algorithm to change less rapidly, b) reviewing actual action and proposals delays in the scenario and/or c) including better story controller into project Emohawk.
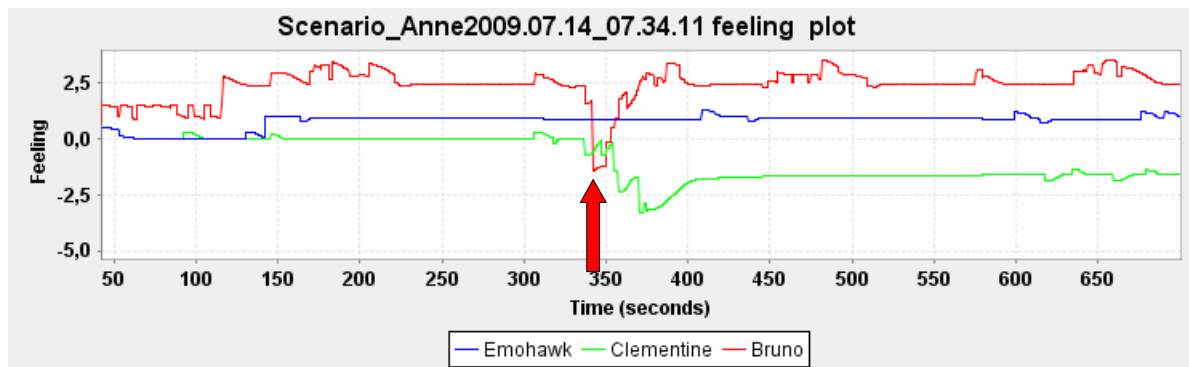
Graph 20. Anne's feeling development. Anne's feeling development in experiment 2009.07.14_07.34.11. We can observe Anne's conflict and feeling recovery toward Bruno around spot marked by a red arrow.

7.4.3 Asymmetric relation experiments analysis

Strong asymmetric relations occurred in 5,8% of all the experiments. One of our concerns was to analyze these experiments as we do not consider asymmetric relations as believable. Two of the most common reasons for these asymmetric relations will be shown on two graphs below (other reason for this to occur was the message loss problem). These graphs will feature the feeling development for experiments featuring asymmetric relations at the end. The description is provided in the graphs comments.
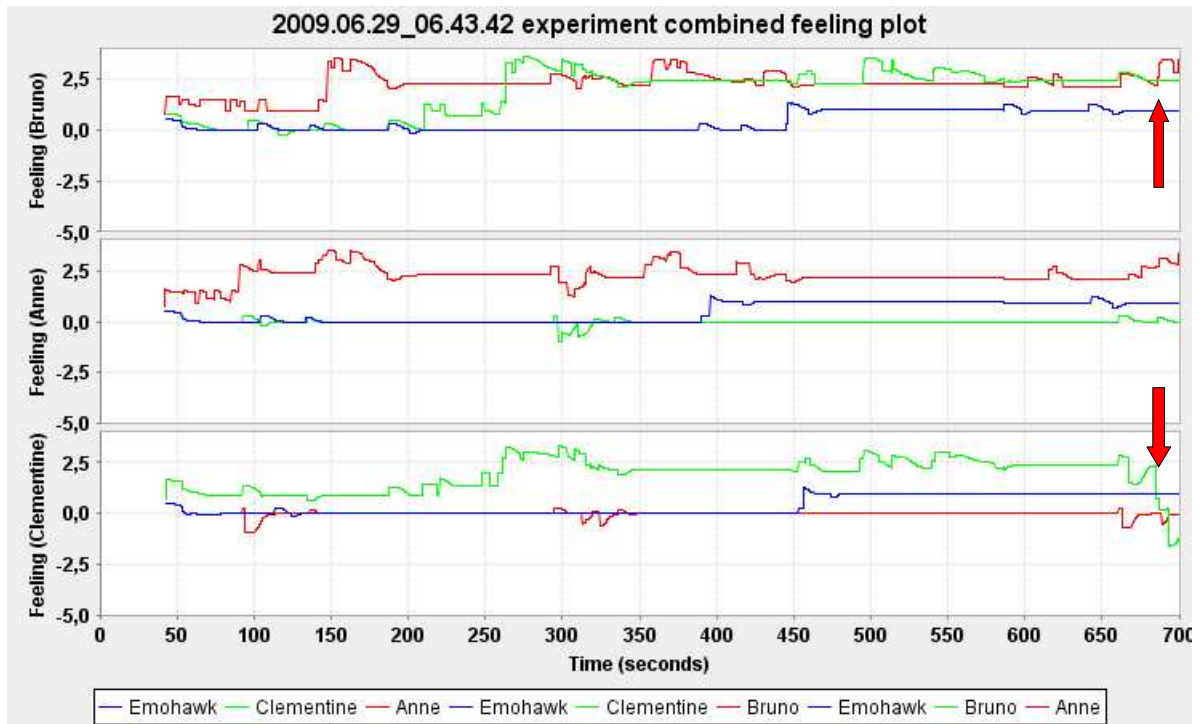
Graph 21. Combined feeling development. Here we see asymmetric relation between Bruno and Anne (marked by red arrows – the green lines). This was caused by Clementine observing positive actions between Bruno and Anne at the end of experiment, but she did not manage to react due to the fact experiment ended right away.
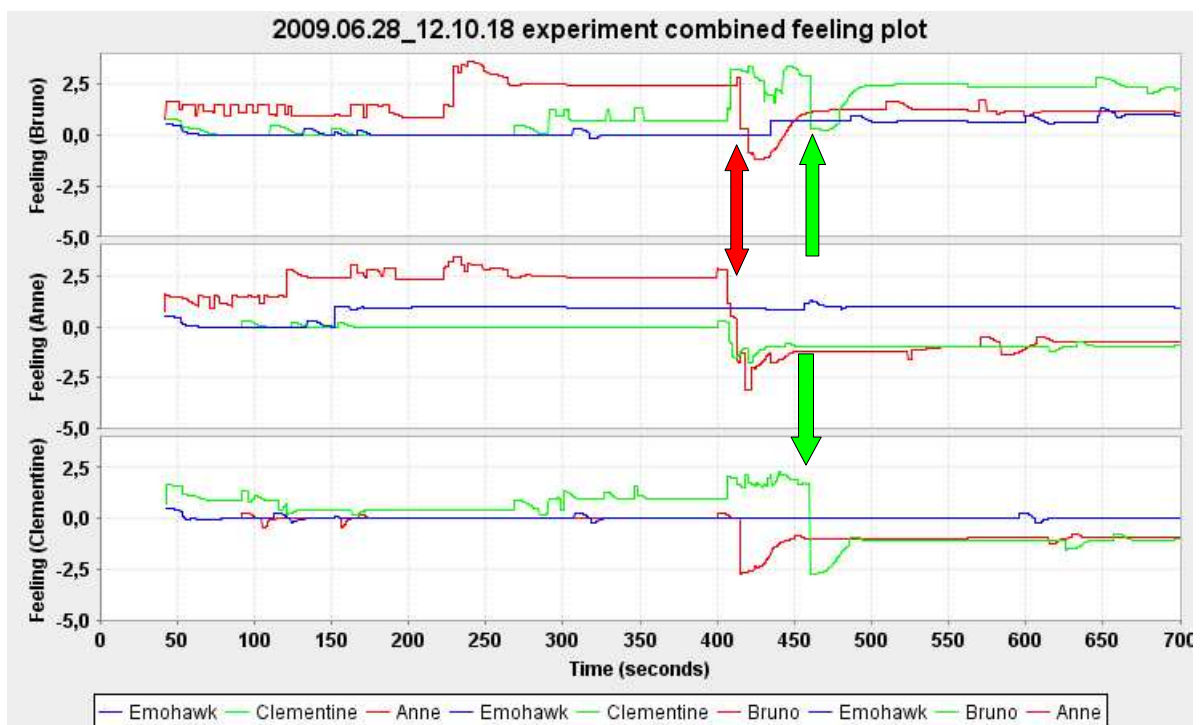
Graph 22. Combined feeling development. We can see conflicts occurred between girls and Bruno in the middle of the experiment (marked by red arrows for Anne and green arrows for Clementine). Bruno managed to recover his feeling value from these conflicts, while girls managed not to. This shows some asymmetries in event appraisal that should be revised.

89

7.4.4   An unintended girl's behavior feature

On some of the girls state graph we have noticed strange pattern to occur (marked by red ellipses on Graph 23). Upon investigating this feature we have found out that it is caused by not resetting one of the girl state variable at the proper time, when certain conditions in the experiment occur. Sometimes this issue caused the girls to remain at home at the second part of the experiment. This probably lowered the number of conflict occurrences in the experiments. This feature affected around 27% of all of the experiments.

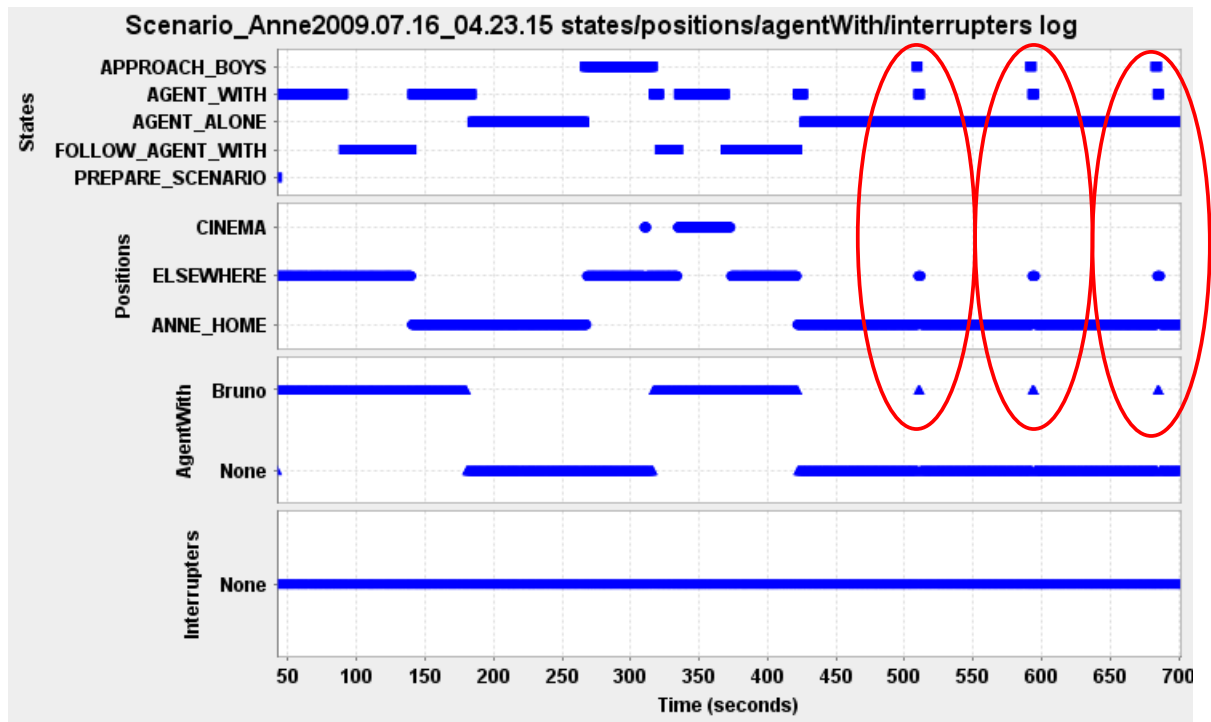This discovery shows our methodology can be used also for testing purposes. Searching regular patterns on the state graph may uncover hidden, emergent affect-architecture features.

Graph 23. Anne's state diagram. Anne's state diagram showing strange patters marked by red ellipses caused by the issue in Anne's control logic.

7.4.5   Failure in path finding/obstacle avoidance code

In 3.5% of experiments our IVAs path finding and obstacle avoidance code failed causing two of our agents – Bruno and Anne to "stuck". This caused them to remain at one place for several minutes until they finally managed to resume the movement.

The path finding and obstacle avoidance code of Pogamut 3 platform is still in development. Its simple beta implementation will be improved in future releases solving the issue. Below on Graph 24 we can observe how the feeling development graph looks in these broken experiments. The interesting parts of the feeling development are shifted to the right (due to the fact usually nothing happens when agents are stuck).
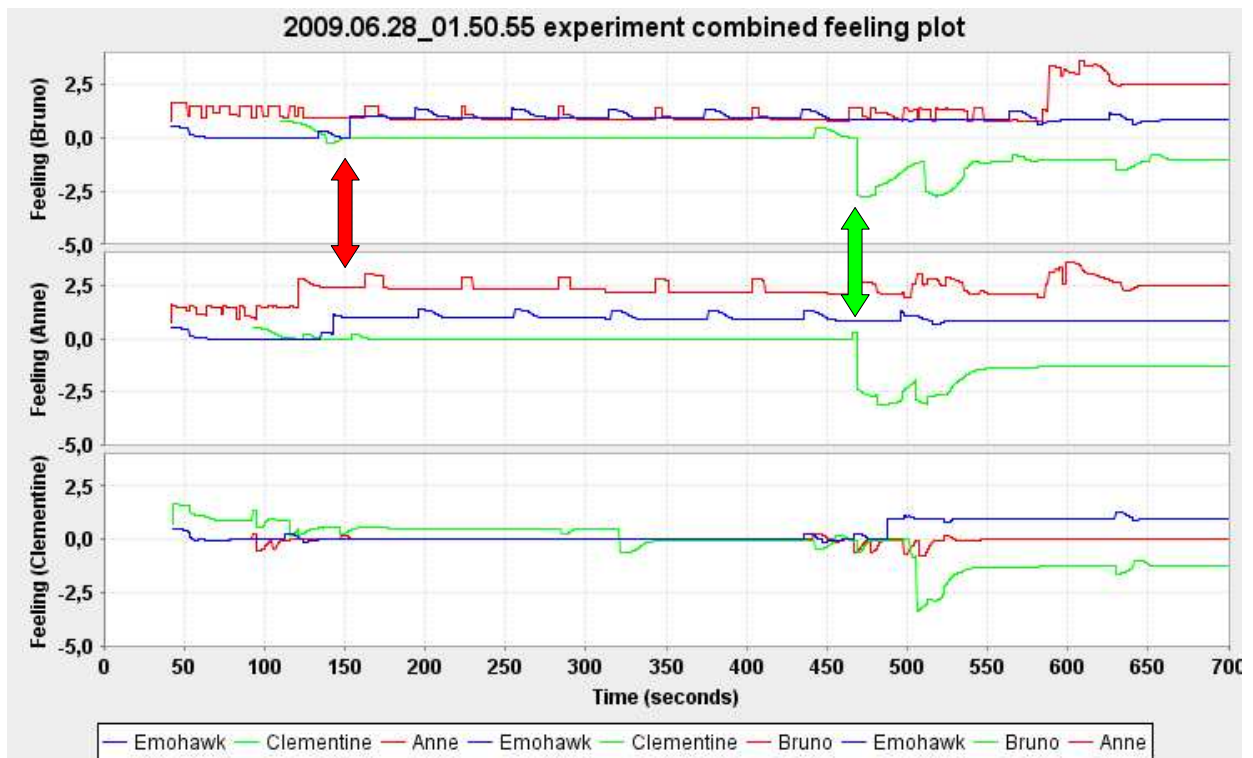
Graph 24. Broken path-finding code feeling development graph. Bruno and Anne got stuck at the time marked by a red arrow. At time marked by green arrow they resumed the movement. At this time Clementine arrived and interrupted our agents, effectively managing to unstuck them.

7.5   Summary

In this chapter we evaluated our experiments regarding the emergent story performance and the believability. We found out that we have succeeded in creation of a system that is capable of producing an emergent story unfolding. Our story featured distinct story ends regarding the end feeling relations between the characters. The story unfolding differed as well featuring stories, where certain key points occurred or did not occur (e.g. intimate action occurrence in the scenario or Clementine not being taken to cinema). More than 80% of all of the experiments featured a conflict – a decrease in the feeling value for at least one character toward at least one other character below minus one. This is a desirable result achieved by random encounters and interaction between characters in the scenario.

Concerning the believability of our scenario we achieved reasonable results for our prototype implementation. The feeling development after emotion events showed strong trends for positive and negative events to behave accordingly with small exceptions that can be easily fixed – although of course message loss problem prevented us from capturing all the events correctly. The numbers of suspicious action and sub-scene pairs were rather small (occurring in less than 5% of all the experiments), but showed some features of our scenario that can be further improved. We have also encountered an undesirable feature connected with the interaction between three and more agents. The problems will be summarized below.

Our analysis based on our methodology succeeded in capturing some problems in our scenario. Here we will go through all of them describing their reasons and sketching the possible solutions.

Message loss problem. Pogamut 3 core beta implementation features an undesirable issue –

there is 10% - 20% loss in the text message communication. On one hand this helped our scenario to produce more emergent story outcomes based on overhearing of certain actions or proposals, on the other hand it prevented our characters from reacting to all of the events occurring in the environment.

We have taken firsts steps to debug this issue by creating a test case that confirmed this issue outside of the project Emohawk implementation.

Agent's navigation problem. We use a simple implementation of path finding and obstacle avoidance in project Emohawk, since this code is still not ready in Pogamut 3 implementation. Our implementation caused agents to stuck in 3,5% of all the experiments for several minutes. After that, the agents were usually able to resume the movement and continue with the scenario.

Solution to this issue is to provide better, more complex implementation for path finding and obstacle avoidance for our IVAs.

An unintended feature of girl's behavior. Upon inspecting some strange patterns in some of the girls state diagrams, we have found out there is a small issue affecting girl's decision making, that is causing a situation, where the girl will remain at home in the second part of the experiment. Clearing an internal girl variable at the correct time can solve this issue.

This issue affected 27% of all the experiments, decreasing the chance of the interaction of all three characters in the second part of the experiments. Still conflicts occurred in more than 80% of all the experiments.

Double interruption problem. When three characters are interacting with each other a situation occurred where two of them wish to end the interaction between each other, but they want to continue interaction with the third character. This causes our two characters to remain on place interacting with the third character, which can later lead to that, they will resume interaction between each other.

When interacting with a group of characters our characters should either end the interaction with all of them at once, or when wishing to end the communication just with some of them, he/she should ask the others he/she wants to remain with to do the same or walk away alone instead.

Feeling based problems. Considering the feeling development, we have encountered two feeling based problems – a) *asymmetric relations problem* and b) *feeling recovery problem*. These problems are partly connected to each other. Sometimes after a negative action takes place in the scenario, the feeling decreases significantly, but after some time, it returns to its original value. This is a problem of event appraisal of some events that are not leading to occurrence of emotions, which create a solid part of the feeling value.

In other situations the feeling decreases, but forth coming positive events cause the feeling to return to positive values rapidly. Now the question here is: Is this the way we want our feeling to behave? We believe that the answer is a rather "no". First, the negative events should be at least partly "remembered" in the feeling value even after the emotions decay. Second, the problem of feeling value returning from negative values to high positive values should be carefully revised as well. At least it should take more time than for example 10 seconds.

This leads us to an assumption that we should revise our feeling computation algorithm and appraisal of some negative emotion events.

We would like to note, that these feeling problems had rather small effect upon occurrences of suspicious or non-believable actions pairs.

Sub-scenes generation and analysis problems. Our algorithm that generates sub-scenes should be modified by not allowing to generate two same sub-scenes in a row.

Other problems are agent to agent interactions that last less than five seconds. In these cases appropriate sub-scene will not be generated. This may cause suspicious sub-scenes pairs to occur in the resulting sub-scene series. To solve this we can either control the agent to agent interactions to last more than five seconds or revise the sub-scene generation algorithm.

Our matching of sub-scene series should be further improved to allow for small errors in matching.

The methodology used in this chapter proved to be useful in scenario analysis. However the number of analyzed experiments was rather high. We estimate that we have analyzed (at least briefly) around 30% of all the experiments because of some suspicious aspects occurred in them. The reasons for this are a) our inexperience in evaluating what should receive greater attention and what should not and b) shortcomings of our methodology, which will be discussed in following chapter.

Overall we treat our prototype implementation of project Emohawk as a success. However to there are still many things to about project Emohawk educational toolkit. We will discuss these things in the following chapter.

8 Discussion, future work

Here we would like first to mention problems connected with measuring and evaluating of believability. In project Emohawk IVAs are often performing actions to each other. Among others, action compliment is chosen quite frequently. Very often action pair compliment-compliment appears in our scenario – as could be seen in the results. Is this believable? Or would this be considered by an average user of our platform as believable behavior? Perhaps not. We have not considered this as a problem due to the fact, that we have designed our IVAs to use this action frequently – to promote positive feeling shifts among them. Well this is perfectly "ok", as long as it will be just us, who will use the platform. To make our prototype believable to an average user, some work remains to be done (we will discuss this further in this chapter).

Our believability analysis was currently based on the things that were not programmed by us and are not desirable – for example as action pair featuring negative and then positive action – this may indicate rather undesirable behavior of our model. Also we have some idea how the feeling should be developing during the scenario – it surely should not be oscillating rapidly between positive and negative values. To summarize, the outcome of our analysis is not that our prototype implementation produces perfectly believable behavior. Instead the outcome is that our prototype is believable as far as we are concerned with believability issues right now. This means our prototype features acceptable behavior with rather low number of abnormalities (we do not count abnormalities included by design). Nevertheless our methodology proved its capability of capturing abnormal prototype behavior. So after we will have more requirements on our implementation, we can use it effectively again.

Overall, we have dedicated large parts of this thesis to the evaluation of the prototype. Was this really necessary at this moment? Perhaps not all of the requirements, we have on the implementation, are known. Truly – it was.

We now have the proof-of-concept – not only the prototype implementation, but also the methodology for evaluation. The evaluation is critical in these kinds of application. Although we would like to note, that we are aware of the fact, that automatic evaluation can never fully substitute the evaluation done by the users of our prototype.

Concerning our evaluation methodology we have to ask ourselves several questions. Is our evaluation methodology effective? Did we actually spared some time by using it? How accurate the evaluation based on it was? Are there any possible future improvements? We start by answering the last question. Summarizing all our knowledge we have got by using current methodology we can think among others of these future improvements:

a) To make the evaluation more automatic. Some parts of the evaluation were done by processing the outcome text files with regexp search. This type of analysis can be done automatically as well – if we know what we are looking for (this may actually be clear just after we see at least some parts of the results).

b) To incorporate automatic analysis of IVAs movement in the environment and extend the feeling development analysis capabilities.

c) To produce multi-graphs that better capture some aspects of the experiments (e.g. combined event and feeling development graph for each character).

d) To analyze agent to agent interaction in more detail (e.g. pairs of action constructed by character one performed action to character two, after that character two performed a

"response" action to character one)

e) To produce special, more detailed outputs for suspicious experiments.

It is hard to measure how effective our analysis was, but we have to say, that it was able to capture some aspects, that would be hard to find by simple "watching" (e.g. the unintended girl behavior feature). If we would like to view all the experiments ourselves we would require 3570 minutes (59,5 hours) of our time. Moreover we would not be able to see everything as we would be limited by the fact we can observe just one location in the environment at one time. Currently it is hard to estimate how much time we have spent analyzing the experiments, since the development of the methodology and actual analysis went hand in hand. However, it was indeed less than 60 hours. The most time consuming activity is connected with the analysis of suspicious experiments featuring some unwanted aspects. Although even this can be further improved by producing more appropriate outputs. Moreover with increasing experiment count (thousands of experiments), it is not possible to analyze them by "watching".

Some aspects of our analysis methodology proved to be better than others. For example comparing the feeling development during the scenario with sub-scenes feeling differences we can say that overall feeling development provided much better measurements for the scenario performance. Sub-scene feeling differences contained a lot of noise and their usability was limited (although it helped to clarify certain trends in our scenario).

To summarize, we have managed to create a working model of affect-driven architecture that is capable of controlling IVA and at the same time produce emergent story outcomes. We have also conducted first evaluation of the model. However to finish our Emohawk toolkit we need to especially:

a) Define a believable interaction scheme, which will be used by Emohawk IVAs. Especially what actions should our IVAs be able to perform? How fast the interaction between the agents should be? Having these questions answered, we can use the methodology to evaluate our newly built prototypes to reflect these new requirements.

b) Revise the user interaction possibilities. Are we going to define some borders limiting the user in his interaction possibilities? How should we treat "evil" users that will purposefully try to sabotage our scenario? How should our agents react onto a user who will perform positive and then negative actions to IVAs in sequence? Do we need to revise our feeling model to deal with this? The evaluation here will be much harder – we will require some model of user interacting with the environment.

c) Extend our affect-driven architecture to take advantage of other affects provided by emotion model ALMA. Currently our IVAs are using just feeling affect that is computed from ALMA emotions. We are planning to incorporate distinct behavior styles for our IVAs changing according to ALMA mood.

d) Revise the key points of our story. Should we add other places to our UnrealVille town? Should we make any other modifications? How many distinct stories we would like to have? Will we use some planning to direct oncoming more complex stories in our implementation?

At the moment we have some of these questions already answered, but some of them do still require our attention. Except of these rather conceptual problems, we will need to deal with more concrete problems such as repairing the discovered issues in Pogamut 3 and project Emohawk implementation. Not to mention to incorporate more civil appearances for our

IVAs as well with animations that will be used to express the actions (which are currently expressed symbolically through text outputs). The extended animation set is almost ready and will be integrated in the next version of project Emohawk.

These extensions will have more positive effects on believability than the use of plausible emotion model – in virtual application, what user sees is the most important as it is known by all of the major computer games developers (and hence so big enhancements of computer graphics and just minor interests in artificial intelligence in modern games – fortunately this seems to be changing).

Emohawk vision. We would like to dedicate the last part of this chapter to our Emohawk educational toolkit vision. Especially we try to answer the question: How will Emohawk educational toolkit provide the education in the field of IVA development? It will provide the students with 3D environment where certain stories will occur. The nature of these stories should allow for visualization of problems the development of IVAs brings.

For example a simple story may be "Bruno wants to take Anne to the cinema". In this story the agents need to solve path finding and obstacle avoidance problems and follow each other. However maybe we would be better starting with a story "Bruno goes to the cinema alone". Our Emohawk toolkit stories will be divided into categories. Each category will feature different complexity of tasks that need to be accomplished in order to replicate the story by the student. We also will provide our own solutions to these simple problems. Moreover more complex stories allowing user – scenario interaction will be implemented showing the problems in the field of emergent narrative, IVA believability and IVA emotion modeling. The good starting point for this more complex education is to create a model with parameters, which can be changed by the student and outcomes in the virtual world can be seen. This is partially accomplished in our current project Emohawk implementation, although a lot of work remains to be done.

We hope we will succeed in designing Emohawk toolkit in a way that offers the possibility to educate students in the field of IVA development in interesting and efficient way.

9 Conclusion

In this work we presented our vision of creating the educational toolkit Emohawk and performed first steps towards it by implementing a prototype version – project Emohawk. This prototype was built upon Pogamut 3 platform using virtual environment of action game Unreal Tournamet 2004, which was modified to feature civil non-violent IVAs. These IVAs are controlled by affect-driven architecture, which was developed by us for this purpose. This architecture produces emergent story unfolding and interaction. We have evaluated this prototype by our newly created methodology in the means of believability and emergence.

The evaluation showed that we have succeeded in creating a scenario that is fairly believable – regarding our definition – and produces emergent interactions as well as emergent story outcomes. However, we have also discovered shortcoming of some parts of our prototype. Among others, we have encountered issues in Pogamut 3 beta implementation as well as some undesirable features of our affect-driven architecture.

To finish the work on Emohawk toolkit a lot of work remains to be done. We need to a) perform revision of the design based on the outcomes of this thesis, b) finish the final implementation, that should incorporate more stories and have better control of user-scenario interaction and c) solve technical problems such as debug the issues we have discovered or extend the virtual environment with more civil models for IVAs and plausible animations of actions. Especially considering b) we will need to incorporate more elaborate way how to set up and control stories in our scenarios (we are considering to use classical planning to solve this problem).

10   Literature:

(Adobbati et al., 2001)
Adobbati, R., Marshall, A. N., Scholer, A., Tejada, S.: Gamebots: A 3d virtual world test-bed for multi-agent research. *In: Proceedings of the 2nd Int. Workshop on Infrastructure for Agents*, MAS, and Scalable MAS, Montreal, Canada. (2001) URL: http://www.planetunreal.com/gamebots [27.7. 2009]

(Alice, 1995 – 2009)
Pausch, R. (head), Burnette, T., Capeheart, A.,C., Conway, M., Cosgrove, D., DeLine, R., Durbin, J., Gossweiler, R., Koga, S., White, J.: Alice: Rapid Prototyping System for Virtual Reality, IEEE Computer Graphics and Applications. (1995-2009) URL: http://www.alice.org/ [27.7.2009]

(AMIS)
AMIS – Artificial Minds for Intelligent Systems. URL: http://artemis.ms.mff.cuni.cz/ [27.7.2009]

(Aylett et al., 2005)
Aylett, R.,S,. Louchart,S, Dias, J, Paiva, A and Vala, M.: FearNot! - an experiment in emergent narrative. *In: Proceedings of IVA 2005*, LNAI 3661, Springer , pp. 305-316. (2005) URL: http://www.macs.hw.ac.uk/EcircusWeb/ [27.7.2009]

(Bevacqua et al., 2007a)
Bevacqua, E., Mancini, M., Niewiadomski, R., & Pelachaud, C.: An expressive eca showing complex emotions. *In: Proceedings of AISB'07*. Newcastle University, Newcastle upon Tyne, UK. (2007)

(Bevacqua et al., 2007b)
Bevacqua, E., Heylen, D., Pelachaud, C., & Tellier, M.: Facial feedback signals for ecas. *In: Proceedings of AISB'07*. Newcastle University, Newcastle upon Tyne, UK. (2007)

(Bida & Brom, 2007)
Bída, M., Brom, C.: Emotion virtual agents. *In: Proceedings of Kognice 2007*. Olomouc, Czech Republic. In print. (in Czech) (2007)

(Brom et al., 2009)
Brom, C., Bída, M., Gemrot, J., Kadlec, R., Plch., T.: Emohawk: Learning Virtual Characters by Doing. *In: Proceeding of TIDSE 2009*. (2009) (in review)

(Cavazza et al., 2007)
Cavazza, M., Lugrin, J.,L., Pizzi, D. and Charles, F.: Madame bovary on the holodeck: immersive interactive storytelling. In: Lienhart, Rainer, Prasad, Anand R., Hanjalic, Alan, Choi, Sunghyun, Bailey, Brian P. and Sebe, Nicu (eds.) *Proceedings of the 15th International Conference on Multimedia 2007*, September 24-29, Augsburg, Germany. pp. 651-660. (2007)

(Champandard, 2003)
Champandard, A.J.: *AI Game Development: Synthetic Creatures with Learning and Reactive Behaviors*, USA, New Riders. (2003)

(Damasio, 1994)
Damasio, A.: *Descartes' Error*, G.P. Putnam's Sons, New York. (1994)

(EA, 2009)

Electronic Arts inc.: The Sims 3. (2009) URL: http://www.thesims3.com [27.7.2009]

(Epic, 2004)

Epic Games: Unreal Tournament 2004. (2004) URL: http://www.unreal.com [27.7.2009]

(Gebhard, 2005)

Gebhard P.: ALMA - A Layered Model of Affect. In: *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'05)*, Utrecht, pp. 29-36. (2005) URL: http://www.dfki.de/~gebhard/alma/index.html [27.7.2009]

(GraphViz, 2009)

GraphViz: Graph Visualization Software. (2009) URL: http://www.graphviz.org/ [27.7.2009]

(Hall et al., 2006)

Hall, L., Vala, M., Hall, M., Webster, M., Woods, S., Gordon, A. and Aylett, R.: FearNot's appearance: Reflecting Children's Expectations and Perspectives. In: Gratch, J; Young, M; Aylett, R; Ballin, D. and Olivier, P,(eds) *6th International Conference, IVA 2006*, Springer, LNAI 4133, pp. 407-419. (2006)

(HUMAINE wiki, 2009)

HUMAINE: Wiki webpage definitions. (2009) URL: http://emotion-research.net/wiki/Glossary [27.7.2009]

(JFreeChart, 2005-2009)

JfreeChart – a free Java chart library. (2005-2009) URL: http://www.jfree.org/jfreechart/ [27.7.2009]

(Kadlec et al., 2007)

Kadlec, R., Gemrot, J., Burkert, O., Bída, M., Havlíček, Brom, C.: POGAMUT 2 – A platform for fast development of virtual agents behaviour. In: *Proceedings of CGAMES 07,* La Rochelle, France. (2007) URL: http://artemis.ms.mff.cuni.cz/pogamut/ [27.7.2009]

(Louchart et al., 2008)

Louchart, S; Swartjes, I; Kriegel, M. and Aylett, R.: Purposeful Authoring for Emergent Narrative. In: *Proceedings, 1st International Joint Conference on Interactive Digital Storytelling ICIDS 2008,* LNCS 5334, Springer, pp. 273-284. (2008)

(Loyall, 1997)

Loyall, A., B.: *Believable Agents.* Ph.D. thesis, Tech report CMU-CS-97-123, Carnegie Mellon University. (1997)

(Loyall & Bates, 1991)

Loyall, A., B. and Bates, J.: *Hap: A Reactive, Adaptive Architecture for Agents.* Tech Report CMU-CS- 91-147. Carnegie Mellon University. (1991)

(Mateas & Stern, 2003)

Mateas, M. and Stern, A.: Façade: An Experiment in Building a Fully-Realized Interactive Drama. *In: Proceedings of Game Developers Conference*, Game Design track. (2003)

(McCrae & John, 1992)

McCrae R., R. and John O., P.: An introduction to the fivefactor model and its implications. *In: Journal of Personality*, vol. 60, pp. 171-215. (1992)

(McKee, 1997)

McKee, R.: *Story: Substance, Structure, Style, and the Principles of Screenwriting*. NY: HarperCollins. (1997)

(Mehrabian, 1996)

Mehrabian A.: Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, vol. 14, pp. 261-292. (1996)

(Ortony et al., 1988)

Ortony, A., Clore, G., L., Collins, A.: *The cognitive structure of emotions*. Cambridge University Press, Cambridge, UK. (1988)

(Plutchik, 2002)

Plutchik, R.: *Emotions and Life: Perspectives from Psychology, Biology and Evolution*. American Psychological Association. (2002)

(Pogamut, 2009)

AMIS: Pogamut. (2009) URL: http://artemis.ms.mff.cuni.cz/pogamut [27.7.2009]

(Rabin, 2002 – 2008)

Rabin, S.: *AI Game Programming Wisdom I – IV*. Charles River Media. (2002, 2003, 2006, 2008)

(Sobral et al., 2003)

Sobral, D., Machado, I. and Paiva, A.: Managing authorship in plot conduction. *In: Virtual Storytelling: Using Virtual Reality Technologies for Storytelling*. Springer. (2003)

(Wilensky, 1999)

Wilensky, U.: NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University. (1999) URL: http://ccl.northwestern.edu/netlogo/ [27.7.2009]

Appendix A – Attachments

1. DVD with:

   - Documentation

     · Project Emohawk user documentation

     · Project Emohawk programming documentation

     · Project Emohawk JavaDoc

   - Project Emohawk implementation (compiled jar file as well as source and configuration files)

   - The outcome files of project Emohawk analysis

   - Project Emohawk Unreal Tournament 2004 extension installation file

   - This master thesis in the PDF format

   - Three videos showing project Emohawk scenario

   - README.TXT with DVD overview