

# Interactive Storytelling: Approaches and Techniques to Achieve Dynamic Stories

Madjid Merabti<sup>1</sup>, Abdennour El Rhalibi<sup>1</sup>, Yuanyuan Shen<sup>1</sup>,  
Jorge Daniel<sup>1</sup>, Alcantara Melendez<sup>1</sup>, and Marc Price<sup>2</sup>

<sup>1</sup> School of Computing and Mathematical Sciences.

Liverpool John Moores University,

Byrom Street, L3 3AF, UK

{M.Merabti, A.ElRhalibi, Y.Shen}@ljmu.ac.uk

<sup>2</sup> BBC Research, Kingswood Warren, Tadworth

Surrey, KT20 6NP, UK

Marc.Price@bbc.co.uk

**Abstract.** In this paper we review different techniques which can be used to achieve interactive storytelling in games, most notably the employment of planning algorithms to decide which and how events should be presented to the player at a given time, and the definition of non playing characters actions and behaviours as a response to the player's actions. Furthermore we will consider game world with more interesting characters that react to other characters actions and behaviour, while forming bonds and relationships with them and the player. We will consider the use of emotional characters to this effect. The incorporation of narrative techniques into the storytelling used in games can help not only to increment the level of interaction between the player and the world and characters, but also to keep the story fluent, thus achieving more realistic narrative. We will examine and discuss some of them, along with some existing approaches and possible applications for interactive storyline generation. We will also review and discuss some areas in which emotional characters have been used to support the player, and in the development of interactive storytelling.

**Keywords:** Interactive Storytelling, Planning, Emotional Model.

## 1 Introduction

Similar to other entertainment media, stories in games play a big role in increasing immersion, building tension and adding interest to the player [1]. However, one main difference from the games to those other media is that games are interactive; they expect participation from the player and in turn, players expect to participate and get involved in the events the game is presenting and the outcomes of those events [2, 3].

Stories can be presented in different ways: they can be linear, branching, parallel or threaded [14]. Games typically follow a linear storyline, where the events of the story are presented in a defined sequence. Nevertheless, it can be argued that making a player follow a defined story can diminish the interactivity level of a game; the player is, after all, following a preset path laid for him [1]. So in order to benefit the most of the story

and still allow the player to feel a high degree of interactivity, the concept of interactive or non linear storytelling has to be introduced. Simply put, interactive storytelling presents the opportunity for players to create their story, to have an input on what is happening in the game world they are placed in, to be the ones who dictate how certain events may come to pass [1, 2, 3, 6]. This in turn will help players to be more immersed in the game and enjoy them in a greater degree, since as stated by Laurel in [4], “story making is a pleasurable activity”, as seen by recent studies[4, 5].

Furthermore, when the player is playing a game, he is presented with a fictional environment which he can explore and interact with; but this world is not desolate, as it is populated by several other characters that can interact with the player. Whether a simple digital opponent as in Pong, a complete town full of villagers as in most Multiplayer Massive Online Games, or the diverse NPCs controlled by a game master in the traditional pen and paper role playing games, those characters are placed in the game world in order to help or challenge the players while they try to achieve a specific goal, or simply to make the game world feel more believable and realistic. Additionally, creating NPCs that can react and adapt their behaviours in response to external factors such as the player’s actions or changes in the game world, will increase the player’s level of immersion with the game itself, as they are more likely to feel identified with the characters’ motivations and troubles [2], thus desiring to continue playing the game and getting more involved with those characters.

Based on this premise, we can appreciate the importance of employing more deep and interesting characters to populate a game world, characters that the player will feel motivated to talk to, rather than simple and apparently void characters with only one line of dialogue and which react only when directly addressed by the player [20]; characters that will demonstrate a ‘genuine’ interest in the events occurring in the game world as they will affect them directly, thus giving a more realistic feeling to the player [22]. As mentioned in [21], “expression of emotion is crucial for increasing believability” and, these interactions may also provide players with a sense of belonging to the game world, as they will be affecting it directly with their own actions and decisions, thus becoming actors in the game world rather than just spectators.

The question now is how to provide the player with an interactive storytelling experience. The most common techniques used, and as noted in [3], are branching storylines which enable the player to select the order of the missions available. However, to achieve this goal, we can employ other techniques, such as parallel paths, threaded stories or dynamic object oriented narrative [14]. On this paper, we will examine and discuss some of them, along with some existing approaches and possible applications for interactive storyline generation. On this paper we will also examine and discuss some current techniques being employed to create emotional NPCs, that is, characters that can simulate emotions and behave according to their current state; we will also mention and review some areas where they can be used and how they can affect the game experience for the player.

The structure of the paper is as follows: in section 2, we propose an overview or planning based interactive storytelling approaches; in section 3, we will introduce emotions controlled interactive storytelling based techniques; in section 4 we will discuss the issues of implementation and future possible direction; before summarising the main points of the paper in the section 5 as conclusions.

## 2 Planning Based Interactive Storytelling

As noted in [3], one example of interactive story telling can be found in a game of Dungeons and Dragons where the person in charge of the narrative, the Dungeon Master (DM), can adapt the way the story unfolds in accordance to the way the players behave and interact with their environment; so in a normal D&D game, whenever the players do something completely unexpected by the DM, he can adapt the story to take into account those events. Of course, the DM may want to stick to the original storyline and he can use several resources, such as NPC's to keep the players in track, but it is the players who ultimately decide how they want to behave and act, and the story moves according to these variations.

What we can see from this example is that we can create interactive storytelling by having a larger entity controlling and adapting the story in accordance to the players' actions. Champagnat et al [6] suggest a narrative framework to simulate this activity. In this framework, they suggest having a set of agents in charge of evaluating the current state of the game, and one director agent, who "chooses a set of relevant actions to be executed" [6].

Another approach to generate interactive storytelling is described by Cai et al [7]; in this approach, a tool based on Goal Net is used to plan the story and Fuzzy Cognitive Maps are employed to analyze the user inputs and decide which path should be followed; they were selected due the fact that they act as "collection of the rules such that it not only concerns the relationships between the causes and effects, but also considers their relationships among the causes". Similar to the work described in Champagnat et al [6], this approach implements an agent, modelled using the Goal Net tool, which will be in charge of presenting the story in accordance to the user actions. In order to accomplish this, a controlling agent keeps information related to the states required to achieve a goal, and the relationships or transitions that connect those states [7].

What makes the Goal Net tool a good fit for creating interactive storytelling is that it can divide goals into simpler states; that way, each scene in the story can be divided into simpler scenes, thus creating several paths that can lead to the conclusion of the defined story; having four different ways of connecting the states (sequence, concurrency, choice and synchronization) allows for flexible and more interesting transitions between the different sates, which leads to the creation of more complex stories. Important to note is that using Goal Net as planning tool instead of Hierarchical Task Networks yields better results in terms of interactive storytelling, since it provides the ability to select scenes on real-time, according to the current context and user input [7].

The engine created by Cai et al consists of a knowledge database, where the scenes and their relations are stored, the fuzzy cognitive goal net engine, a container for the agents to be used, known as the Drama Manager, and a Multi Agent Development Environment (MADE) [8] platform to implement the agents' system; using fuzzy cognitive maps, the engine will decide, at runtime, which scenes from the knowledge database should be loaded into the agents in order to better suit the path selected by the user [7].

Three main advantages can be distinguished from employing this method to create interactive storytelling [7]:

- All events are simplified into less complex scenes
- The engine can react not only to the user inputs but also to the actual state of the game, and look for the best path.
- Since all the events are loaded into the engine on real time, its performance is increased.

While the approach presented by Cai et al addresses the issues of letting players interact with their environments and creates its own story, it only uses defined actions to determine how the story will be told. Ciarlini et al [9] suggest an approach in which player's actions affect not only the scenes of the story, but also the behaviour of non playing characters directly related to those actions. What they propose is to use a "formal logic model" for defining "events and characters' behaviour" [9].

Ciarlini et al state that the user should be able to "intervene in the story" but their actions should not "violate the rules of the intended genre" [9]. The architecture used in order to achieve this is comprised of a plot manager, which will serve as the interaction point for the user and will manage the story as it is being created; an Interactive Plot Generator which will take the input from the user and generate and order possible sequences, which in turn will be passed onto the plot manager; finally, a drama manager will control the characters in the game world in order to add appropriate drama to the story; it will also be in contact with the interactive plot generator to maintain the "coherence between logical and graphical representations" [9].

The Interactive Plot Generator (IPG) uses Prolog files that contain the logic to be used for creating stories; in order to generate the storyline, the IPG requires 3 main items:

- An initial state for all the characters and scenarios. These states are described using Prolog and they are contained in a database. An example, as presented in [9], would be: character (villain, dragon) which will represent that the dragon is a villain.
- The logic rules, which will be used to infer the goals for each character as the plot is generated.
- The list of operations available for the characters. Each operation should indicate which events should happen before it (preconditions), which events will come to pass after the operation (post-conditions), the representation of the operation, that is, the smaller actions involved in the performance of the operation (getting close to the victim, etc.) and the actors involved in the event, whether characters or other features.

When plots are generated, partial ordering is used to arrange the events comprising the story; this will allow the creation of more dynamic and diverse stories, as the outcome of each event will be affected by the order in which they are defined to occur. So in some stories, the hero may be stronger before he faces the enemy, and in some other, it will be the other way around.

For every character in the story, a set of goals will be defined following the inference rules created; these rules employ "meta-predicates to speak about the occurrence of an event or the truth value of a literal at certain time" [9]. For example, to represent the motivation of a hero to rescue a victim that has been kidnapped, the predicate rule will be something like 'whenever there is a villain and a victim, if at time1 it is established

that the villain kidnapped the victim, therefore it exists a time<sub>2</sub> on which the victim should not be kidnapped by the villain and time<sub>1</sub> should be lesser than time<sub>2</sub>'. We can observe here that there's no actual mention of the hero rescuing the victim, but what it is modelled is the motivation; there should be a later time on which the victim is freed. Using this approach, the goal of freeing the victim may be assigned to any suitable character, and not just a specific one.

At the initial state, goals will be defined for each character and a planner will decide which events will allow the characters to achieve their goals. When all goals have been either met or abandoned, a new set of goals will be inferred for the characters, and this process will be affected by the past outcome of past events [9]. What one can appreciate from this approach is that it takes into account all previous events, whether they were successful or not, thus affecting the development of the storyline, a feature that is not supported by conventional plan generators [9].

The approach used by Ciarlini et al shows that mixing player's participation, the generation of plans and the inference of goals can yield positive results. The use of predicate logic to construct the scenes and behaviour of characters in the story increases the level of interaction and the degree of interactivity perceived by the player. It is also implied that this model to construct interactive stories can be used in other entertainment media, such as TV or cinema, as a tool for authors to create adaptive stories, and it may even be applied to other non-gaming related applications, like information systems [9].

As it has been suggested, one can employ non playing characters' actions and behaviours to generate interactive storytelling since it will be them with whom the player will be interacting the most while playing the game. In our personal experience while playing D&D and generating dynamic storylines on the fly, we believe that NPCs are a very useful resources for helping the players shape the story to their own liking while maintaining a sense of coherence on it: they might encourage them to keep going in a certain path, point them into the right direction or simply be used to maintain the story interesting.

Barros and Musse [10] suggest an approach that employs this mechanism and two other narrative practices into a storytelling generator based on the usage of planning algorithms and STRIPS domains. The architecture they propose is divided into 3 modules

- A module who will be dedicated to defining the actions that can be performed by the NPCs.
- A module in charge of executing the available actions for the NPCs and keep the state of the world
- A module responsible of resolving conflicts that may appear whenever a new plan of actions is generated.

In order to progress the story, a plan of actions should be created for the characters to follow; each action is described in a language similar to STRIP and it is composed of a prerequisite, used to decide if the action is coherent given the current state of the world, and an effect, which indicates the repercussions of performing the action. For example, the action of giving a present to someone may have as prerequisites having the present and being at the same place as the receiver, while the effects would be to be more liked by the receiver. The implementation made by Barros and Musse "treats the planning problem as a state space search problem, and uses the A\* algorithm to solve it"[10].

What Barros and Musse incorporate into their model for creating interactive storylines is three narrative principles:

- View story as a whole: By considering the overall storyline as a set of sequential plans, which in turn are sets of sequential actions, one can make sure that each plan will try to include the best actions to take in order to continue with the main plot; this kind of control will yield more coherent and believable stories.
- Three act storylines: Being the most common format for stories, it presents the opportunity to group together events that help develop the story: the first act introduces the overall problem, the second one helps increase tension and develop solutions and the third one presents the resolution of the story [2, 10]
- Avoid narrative stalls: Being the focal point of the story, the player may sometimes decide to not follow the plan set in order to continue the story. To avoid this, the actions of the NPCs can be prioritized, so that they can act and accomplish goals before the player. This will allow the story to be fluent even when the player gets stuck.

By combining the narrative techniques and the described architecture, Barros and Musse [10] were able to create interactive storylines that would adapt to the player's actions, even if those actions were not doing well in keeping the story fluent. One possible drawback might be the use of "predicate logic to represent the world state" [10] due the fact that it does not helps us to deal with the most complex aspects of narrating stories [10, 11]; however, as stated by Barros and Musse, it can help us understand the problems and their solutions in an easier manner.

What can be definitely useful for future work is the employment of the narrative principles to create more interactive stories; by defining both the actions and behaviour for the NPCs as a results of the player's actions, will help increase the interactivity of the story: the state of the world and their inhabitants will change to adapt to whatever path the player decided to follow. Allowing the story to keep going whenever he gets stuck can be viewed as both beneficial and detrimental for the player: on one side, the story is still flowing, even when the player cannot find a solution to achieve a certain goal; on the other hand, events performed by the NPCs in order to advance into he story may not be noticed by the player, which in turn can later lead to confusion [10].

Similar to the approach used in [10], the Storytron engine [15] employs a set of verbs to define all the possible actions and it also describes the conditions and consequences for that action. However, and differently from the approaches mentioned early, the creation of plans to be executed by the characters is not handled by a single planning manager, but instead by the characters themselves; that way, each character will decide how to react to a given event.

Plans are created in a 'Reaction Cycle', which allows the characters to examine the event that is occurring and then evaluate the better course of action for them in order to respond to that even; and then executed in an 'Action Cycle' in which the characters will evaluate if all the requirements are met for the execution of their plan and if possible, execute the planned actions. For example, if a character punches another, all witnesses of that event will analyze their possible options (i.e. run away, hide) and will select the one to which they feel most inclined and immediately create a plan to perform that action whenever all the conditions are met.

By employing verbs to define actions, the Storytron engine provides an approach to generate interactive storytelling that implements the strong points of branching narratives, but without the complex managing of the nodes created in that kind of technique [15, 14]. The very interesting feature is the implementation of the Response Cycle, as the idea to allow each character to generate its own plan in response to the events happening in the world allows for more freedom and dynamic interaction of the players, thus providing the sense of interactive storytelling as the events presented will be as responses to the actions of not only the payer, but the whole lot of characters.

### 3 Emotions Controlled Interactive Storytelling

On this section we will examine and discuss some current techniques being employed to create emotional NPCs. We will also mention and review some areas where they can be used and how they can affect the game experience for the player.

#### 3.1 Modelling Emotions

As the player explores and travels the world he is placed in, he will eventually have to interact with the different NPCs that populate the world; furthermore his actions and behaviours will also affect the world he lives in and indirectly, the lives of the other characters in the game. Placing the player in a situation where their actions and behaviours have a lasting effect on the characters he interacts with can be a very rewarding experience [12]; so the main issue is how to model the non-playing characters behaviours in a way that they react accordingly to their emotions, the environment and their changes.

One approach described in [23] is to use a simple flag system to define the current affective states of a given character in response to actions by the player; this approach however results are very simplistic and not very realistic: after all, human behaviours and feelings are related to a more wide range of variables. However, trying to represent all emotions present in humans within one game character might be a nearly impossible task, especially since, as mentioned by [24, p 377], the term might sometimes be used to refer not only to feelings but also to biological needs such as thirst, hunger or sleepiness.

To deal with this issue, Sheldon [20] presents an approach on which each character has a personality chart containing individual scores for six different areas: Love, Admiration, Trust, Loyalty, Respect, Like. Each score can take a predefined value ranging from negative to positive six; the addition of those values will determine the overall character's personality towards the rest of the world inhabitants. In order to determine the relation between the player and a given character, we need to calculate a relationship modifier value, which ranges from negative to positive six and might increment or decrement as a result of the actions of the player, for example, completing specific quests. The relation score between a player and a certain character is then calculated as the addition of both the player's modifiers and the character's personality scores; this score will be compared with a behaviour table to define the attitude for the character towards the player [20].

Another approach is the one presented in [23], which consists of an affinity system that indicates how comfortable a certain character feels towards another character or object. The different situations being presented over the course of the game will increase or reduce the affinity score by a defined amount; a set of rules should be established to define, based on this affinity score, which behaviour should a certain character present when interacting with the player.

### 3.2 Motivations and Social Relationships

While these approaches address the relationships between players and NPCs and help construct more realistic interactions between them, they do not consider other factors that might affect the relationships between characters, such as personal motivations or social networks; these factors are something that should also be considered when designing reacting and emotional characters, as the actions that those characters might be able to do or the behaviour they could exhibit and the consequences of the player's actions might be completely different depending on the social network that a character belongs to [20].

Bailie-De Byl, in [24], presents and reviews some existing architectures oriented towards the creation of emotional NPCs while taking into consideration their inner motivations and desires:

- **Blumberg's Silas T Dog:** On this architecture, an artificial agent is composed of three main layers: Geometry system, Motor Controller and Behaviour System. The first two are in charge of the movement of the agent in a 3D environment, while the latter is the one that handles the responses the agent exhibits to external stimuli; it consists of a hierarchical network of modules, each representing a defined goal which can be either general or specific. When a stimulus is received by the agent, the Behaviour System analyzes that stimuli and employs the motivation and goal variables from the agent in order to determine which behaviour module should be activated. If the selected behaviour has any child behaviours in the hierarchy, all those children modules begin competing between themselves to define the one that should take control of the agent at a given moment; this competition is made by analyzing the internal goals, motivation and the behaviour's importance. Once selected, the behaviour module will inform the Motor Controller and Geometry system of the actions that the agent should perform in order to reflect the change in attitude.

**PETEEI (A PET with Evolving Emotional Intelligence):** Composed of three main process models (emotional, cognitive and learning), this architecture waits for an event to occur and then processes this stimuli to define its desirability. The agent then takes into account its own motivations and processes both internal and external stimulus to define the emotion that should surface. Once the emotion has been defined, the agent evaluates, by employing a set of defined rules, the behaviour that should be exhibited. One interesting aspect that the architecture employs is the fact that the emotions decay over time and the internal variables are evaluated and updated after a fixed time, in order to define if a change on the agent's current behaviour should be made.

**Emotion-Based Control (EBC) Framework for Autonomous Agent:** Whenever the agent receives an external stimulus, a Perception System is activated to determine its

emotional significance, which will affect the resulting behaviour used by the agent; this data is then sent to the Emotion System and the Behaviour System, which also receive internal information from a Drive System in charge of storing all information regarding the agent's motivations and desires. After analyzing the provided information, the Behaviour System, using affect programs (internal pre-defined systems that describe and represent primary emotions), decides which behaviour the agent should present and instructs the Motor System to act accordingly. Something of note is that each affect program has a decay value that represents the time that the actual emotion will be activated; another interesting feature is the possibility to actually combine the basic emotions in order to simulate more complex behaviours, i.e. combine fear and joy to create guilt.

**Emotionally Motivated Artificial Intelligence:** Whenever an agent receives an external stimulus, a first mechanism activates the drivers controlling the motivations for the agent; these drivers describe the source and current strength of all the agent's motivations and will serve as indicators of the behaviour the agent should take. The drivers will remain active until a certain threshold is reached, meaning that a necessity has been covered. Higher emotions are represented in this architecture as coordinate points in a six-dimensional space, known as the affective space; based on this coordinate system, the agent can assess their importance and impact to decide which behaviour should be shown. One aspect of this architecture is that all external stimuli is positioned in the affective space and then processed, along with the motivations of the agent; this means that the resulting behaviour will not only be dependant on the external factors and motivations, but also in the current affective state of the agent.

As we have discussed, these architectures try to create deeper characters in terms of the emotions and behaviours they display by taking into account not only the external factors but the agent's motivations as well, but we still have to take into account the effect that social networks existing between the NPCs and the character itself might render into their interactions.

Lankoski and Björk [13] mention that humans, being social by nature, feel more compelled to get involved with games when social relationships exist between the characters populating the world [13]. Social relationships are an important aspect to be considered when trying to create realistic worlds and societies within it. Even the most emotional accurate characters will feel somewhat unreal if they do not seem to have an interest on the world they are living on or if they do not interact directly with other characters living in the same place [22]. As [25, pp 1] explains it, "to create believable agents we must not only have some internal model of emotion, but also a way to communicate with the user based on these models".

As with human societies, the proximity of NPCs between one another and to the player, their frequent interaction and their personal goals and motivations will result in the creation of relationships between those characters and the formation of groups or factions that share interests or goals [20, 24]. Depending on the existing relations that a player has with different NPCs and the diverse factions that exist in the world, he might gain more or less information, obtain a varying degree of rewards or be faced with different challenges; we have seen this in games such as *The Elder Scrolls: Morrowind* or *Fable*, where the player's actions and allegiance directly affect the behavior and responses he gets from NPCs and the events that are presented to him. However, the modification of those relationships are normally done as scripted events presented in the form of cut-scenes, or by providing the player with a limited set of choices to select

from, thus reducing the input of the player and limiting the number of possible reactions and consequences to just a handful of pre-defined results [13].

So what we now must address is how we can generate and maintain those relationships, both between the NPCs and the player and between the NPCs themselves. Lankoski and Björk present two methods to analyze social networks between characters and to apply them into the game:

- **Social Network Analysis:** This approach represents people as the nodes of a graph and their relationships are represented as edges; since this approach is based on graph models, we can employ mathematical measures to represent diverse characteristics of the relationships. Some examples of the subgroups that we might find within networks are N-cliques (everyone in a group is connected to the rest of the nodes by at most N relationships), K-plexes (members are related to each other node with the exception of K nodes), isolated communities (star shaped groups where each member is related to each other through one central member) and singletons (members with no relationships to the groups) [13, 26]. This kind of analysis can help us to describe how a character fits within one group and how they perceive and are perceived by other members of that group, thus helping us understand why they exhibit certain behavior [26].
- **Actor Network Theory:** This theory states that nodes or members in social relationships should not only be defined as humans or characters, but rather as “collections of heterogeneous entities consisting of humans, human-tool combinations and non humans (e.g. technologies, machines or materials)” [13, pp. 2]. Basically, actor network theory suggests that relationships and behaviours of human actors in a network are defined by other external factors such as experience, existing regulations or a given tool’s capacity and characteristics [27]. One approach that Actor Network Theory employs is the consideration that social groups are not static entities and they are changing constantly, meaning that goals and motivations for the group and its internal actors might be changing constantly; therefore, NPCs and players themselves will have to adapt their actions and behaviours constantly in order to satisfy their new needs and achieve new goals, thus creating more realistic and interesting social groups [13].

### 3.3 Cooperation between Characters

Aside from interaction between characters, social networks are defined by each individual’s goals and motivations [25], which in turn can lead to either cooperation or conflict between NPCs and the players [13]. This presents a more realistic approach to the representation of social groups and character behaviours, as relations between NPCs seem more real when each of them follow their own motivations and the cooperation and social relationships appear as a result of this fact [28]. After all, each character is a single entity looking to satisfy its own goals and necessities; working together with other characters might help them to faster satisfy their needs.

Studies have been made applying the Iterated Prisoner’s Dilemma to model cooperation between agents and satisfaction of each agent’s own needs [12, 29]. Cohen et al point out that the IPD can be used to represent a real social life problem: “how to sustain over time a pattern of cooperation between agents that may be quite beneficial to both” [29]; in their study, they point out that a number of playing agents using the IPD

behave and present characteristics similar to human social networks: groups are formed and cooperation within them exists to varying degrees, always changing according to each agent's own motivations and the current member of that group.

One interesting aspect of the study by Cohen et al [29] is that they allowed agents to change strategies over time as a result of their interaction with other agents; that way each agent would try to adapt himself to the strategy that better allowed him to satisfy his own needs while still cooperating with the rest of the group. To represent this change in strategies, Cohen et al make use of probability values for the initial movement of the agent when no history is present, the probability to cooperate after other has cooperated and the probability of cooperation after other has defected; they also included a noisy environment in which sometimes the messages between agents were not perceived correctly, which helped create more realistic interactions between agents as "real interactions may always involve failures to execute what was intended, misperceptions of the actions of others, misjudgements of the experience of others and errors in understanding and emulating the strategies of others"[29, pp. 6].

The obtained results showed that agents tend to be more cooperative when they were able to adapt and change their strategy after some interactions with other agents; they also showed that populations and groups managed to 'correct' themselves, that is, balance the level of cooperation from too much to too little. These results would be supportive on the idea that agents, or characters, and the relationships existing between them change accordingly to their interactions, their existing social relationships and the environment they are placed on.

An alternative approach to generate collaboration between agents proposed by Puustinen & Pasanen is the employment of a Nash equilibrium [29], which is "a vector of action selection strategies, in which no agent can unilaterally change its strategy and get more utility"[28, pp. 1]. The premise behind this approach is that every single agent is always trying to maximize its own benefit, even if that means sometimes acting against the rest of the agents, while also trying to find a Nash equilibrium. In order to do so, each agent employs a utility function to evaluate which action given the current group's actions will yield the most utility. Each goal is described as a term or perception and has an assigned importance value; each round, every agent's utility function will verify the current state of the game and, by employing the weights of each term, calculate the resulting utility of possible strategies to take.

Each agent's strategy is recorded by a controlling module, which is in charge of verifying that the Nash equilibrium has been reached; if that is not the case, the agents are informed and they define another strategy. Once the Nash equilibrium has been found, each agent is informed and they proceed to execute their own strategies, which is surely to render the best utility. We could say that the controlling module acts in a similar way to what a chess player or dungeon master does when moving pieces across the board or controlling NPCs during battle: they analyze the possible actions to be performed and then decide which strategy will yield the most benefit; the difference in this case is that each agent will individually provide a possible strategy based on its own goals.

Another benefit of this approach is that the actions performed by the agents appear more realistic, as they are motivated by their own goals and act accordingly, and the group's actions and movements result more coordinated and intelligent, which adds a feeling of believability to the players [28]. One possible problem to take into account when using this approach is the total time it takes for a Nash equilibrium vector to be

found; results found by Puustinen & Pasanen [28] indicate that, if the NPCs' number of possible actions is limited, as a result of providing fewer terms or by the existence of dominated strategies, the total time might be short..

### 3.4 Emotional NPC and Interactive Storytelling

We have reviewed some techniques and methods to create emotional NPCs and the relations between them, and explained how those can be affected by interacting with the player or other characters. As we have discussed, emotional NPCs can be used to add a higher level of realism to the game world as their actions and behaviours will vary depending on the actual state of the game world, their inner emotions and goals and their own social relationships. While emotional NPCs can be used in action or social simulation games (see [12],[28]), they also lend themselves nicely as helpers to create, enrich and unfold stories created using traditional narrative or, more interestingly, interactive storytelling methods and applications.

As Pizzi and Cavazza state, "the psychology and feelings of characters is an essential part of most dramas" [30, pp.1], so their usage of emotional NPCs and their relationships to construct, develop and unfold dynamic stories can be a very helpful technique. We review two approaches that address this idea in two different aspects for story generation.

Narrative can be divided into two main components: actions that can be executed by actors, and dialogues between characters to dramatize some of the performed actions [31]. Based on this premise, the model proposed by Cavazza and Charles focuses on the employment of the character's emotions and social relationships to generate realistic dialogues relevant to the current event being narrated to the player. They do this by implementing an affinity system between characters to indicate their social relationships and emotions towards one another, while a rule based system uses that information to generate the dialogue template to be used.

To test the model, the authors generated a group of authors which were individually being invited by another character to a party; each character would accept or refuse the invitation based on their emotions towards the inviting character, and their answer, either positive or negative would also be affected by the affinity between the characters. The results presented by Cavazza and Charles [31] show that constructing dialogues based on the character's emotions and relationships can result in the creation of more realistic situations and interactions, which could in turn, help develop the storyline in more interesting and non-linear ways.

The approach presented by Pizzi and Cavazza [30] employs character feelings and motivations as the defining variables that indicate how the story is unfolded. On this approach, every character has a defined set of goals and feelings defined as a set of rules in a STRIPS-like language; at a given moment, a planner module will analyze each character's emotions, goals and motivations, along with the current state of the game in order to determine the best action for them [30]. Once actions for each character have been selected, this information will be sent to the graphic engine which will update the game world accordingly.

The authors decided to test the model by employing a passage from the novel *Madame Bovary*, which focuses strongly on the psychology of the characters to tell the story of a bored high-society woman and her pursuit for happiness. Users would be able to have

some input as characters interacting with the main NPCs, in order to affect their emotions and behaviours. The obtained results showed that interactions with the characters and the social relationships they establish can have an input on their behaviours, which in turn affects the actions they perform, thus changing the events presented in the story: one simulation ended with the main character leaving her husband for her lover, satisfying her need for happiness with him while other, on which another character empathized the joy of creating a family, resulted in the character leaving her lover to enjoy and protect her own family [30]. These experiments show the potential offered by emotional NPCs based techniques to support interactive storytelling.

## 4 Discussion and Future Direction

As we have seen with [9, 10, 15], when creating interactive storylines, the player's actions trigger a series of choices regarding which and how events may be presented and how non playing characters act and behave in response to those actions; however, this approach has the drawback of having to model all possible actions and behaviors that can be assigned to any NPC. A solution for this would be to have the NPCs react on their own to their environments, thus allowing them to relate more realistically with the world and their societies, which in turn will elevate the degree of interactivity for the story as, stated by Lankoski and Björk, "social relations in games are typically part of the storyline" [13]; one cannot deny that exploring the relationships between characters can provide better game experiences [4].

To generate more complex characters behaviours, as explained in [14], we could use flag or affinity systems to alter the behaviour of non playing characters as a result of the player's actions. A flag system is just a set of Boolean variables which change from 'yes' or 'no' values depending on the player's input; for example, being rude towards an NPC may prevent that character to help the player later in the game. A more complex approach is the affinity system on which the characters have a certain range of affinity towards each other, increasing and decreasing as a result of the interactions they have; so if a player is rude to an NPC, but later he saves his life, then the NPC may still be willing to help the player out as a result of the interactions they had. This type of relationships has been used in games such as *The Elder Scrolls: Morrowind*, or *Star Ocean: The Second Story*, which explore the relationships between characters and those relations, along with the player's decision's affect the way the story is unfolded.

A more detailed approach to achieve this is proposed by El Rhalibi et al. [12] [18][19] which use a rule based system (RBS) and iterated prisoner dilemma (IPD) to simulate the emotions and interactions of non playing characters. By using the emotional drives based model to manage the character's needs, their behaviours can be changed in order to fulfil those needs; then the RBS is used to decide which actions should be taken in order to satisfy the current needs; the agent's behaviour is then affected by the outcome of those actions [12][18][19]. To model the relationship between agents, the proposed approach uses the IPD to "give a straight forward interaction decision based upon the previous interaction with the agent" [12]. This way, we can obtain several interactions based on the previous actions and behaviour the player

showed towards the NPCs, thus allowing for more realistic social relations between them which in turn can help to unfold the story.

One approach that has not been exploited much is to create storylines following the dynamic object oriented narrative, in which the overall storyline is divided into sets of episodes which in turn are divided into scenes, all of whom may be connected by another simpler technique such as linear or parallel narrative [14]. This approach to generate stories has the advantages of adding versatility to the storyline, as one player can begin the story in several defined points across a range of episodes and decide on its own how to advance through the scenes [14], and simplifying the way scenes are interconnected, while maintaining a global sense of the overall story. Another benefit is that it also can be used to maintain the three act structure, as suggested in [10], as each act can be comprised of several episodes.

Including emotional non playing characters would allow for more interactivity and immersion, as their relationships with the player would affect the way the story is created, as it occurs in the Storytron engine [15]. This feature can also be complemented by defining motivation instead of actions, as described in [9]; that way we can define the same motivation for several characters but the character or characters who will respond to that motivation may vary in response to other factors; finally, NPCs can be used to prevent the player from getting stuck and maintaining the flow of the story.

The implementation of emotional NPCs can be a very helpful resource in interactive storytelling to define how to present the story to players in a more realistic way; based on the methodologies reviewed in this paper, the authors believe that allowing characters to define their own emotional states and relationships based on their goals and needs, along with the implementation of a master control module in charge of examining the game state and deciding how the story events should be presented, can result in a very interesting framework to created dynamic stories.

Emotions and relationships could be modelled using an affinity system similar to the approaches presented in [20] and [23]: each character could employ a control module that, based on their motivations, goals and relationships, defines the current emotional state and behaviour for that character. A global control module would examine the current game state in order to define which event in the story should be presented next. That way, both players and NPCs emotions and motivations help shape the existing relationships between characters and the actions they make, thus affecting which events are presented to the player , generating more realistic dynamic stories. The implementation of such framework presents an interesting opportunity for future work.

A planning manager module can be implemented to analyze the state of each scene right at the beginning and, according to the player's actions, previous events and NPCs current behaviours and states, decide which actions should be included in a plan to keep the story flowing and which possible characters could perform them; this approach would employ techniques and ideas similar to the presented in [6, 7, 9, 10, 12].

As appealing as it may seem, one important thing to note regarding interactive storytelling is that, as pointed out in [14], it adds complexity to the whole development process of a game, as more pre-production and testing must be performed in order to create an enjoyable experience for the user. However, if handled appropriately, the inclusion of interactive storytelling can make a huge difference and add features that broaden the horizons in the field of not only games, but interactive entertainment as well.

## 5 Conclusions

As we have discussed in this paper, there are different techniques which can be used to achieve interactive storytelling in games, most notably the employment of planning algorithms to decide which and how events should be presented to the player at a given time, and the definition of non playing characters actions and behaviours as a response to the player's actions; the incorporation of narrative techniques into the storytelling used in games can help not only to increment the level of interaction between the player and the world and characters, but also to keep the story fluent, thus achieving more realistic narrative.

On this paper we have presented several approaches that can lead to interactive storytelling; however, while all approaches have strong advantages to address the generation of more dynamic stories, they also have some drawbacks to take into account, as some may be disrupted by complex stories [10], others may have it difficult to incorporate a formal set of rules for most modern genres [9] and some require several components to work properly [7].

While not perfect, all these approaches serve as bases to propose an alternative approach to create interactive storytelling, which utilizes the not so used dynamic object oriented narrative to handle the story arcs and employs some of the ideas and techniques presented in this paper in order to create a new framework which benefits from the strong points and tries to minimize the drawbacks of existing implementations.

We have also reviewed some techniques used to create emotional NPCs and to analyze and model the social relationships that can be created when they interact with the player or other characters. We have discussed some techniques to promote cooperation amongst NPCs, so that they can satisfy their own needs and motivations while also maximizing the gain value for the group they belong to.

As we have discussed, the inclusion of emotional NPCs and their social relationships to games can lead to a higher immersion level in the player, as players will get a more realistic feeling when dealing with them, since their actions and behaviours will be motivated directly from their own goals, emotions and past interactions with other characters, thus making them more believable.

While interactive storytelling can be applied most notably to games, there are also other fields where it could be used, such as medical simulations, training or educational software such as [16, 17], which currently may use defined and linear storylines. Introducing interactive storytelling into these types of applications, can increase their level of immersion, thus providing the user with a more enjoyable and attractive experience and a base for systems employing alternative types of interaction with users [21].

## References

1. Rollings, A., Morris, D.: *Game architecture and Design*; The Coriolis Group (2000)
2. Rollings, A., Adams, E.: *On game design*. New Riders (2003)
3. Rouse III., R.: *Game design, theory and practice*. 2nd edn. Wordware Publishing, USA (2005)
4. Laurel, B.: Narrative construction as play. *Interactions* 11(5), 73–74 (2004)

5. Salovaara, A., Johnson, M., Toiskallio, K., Tiitta, S., Turpeinen, M.: Playmakers in multiplayer game communities: their importance and motivations for participation. In: Proceedings of the 2005 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2005), Valencia, Spain, June 15 - 17, 2005, vol. 265. ACM, New York (2005)
6. Champagnat, R., Estraillier, P., Prigent, A.: Adaptive execution of game: unfolding a correct story. In: Proceedings of the 2006 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2006), Hollywood, California, June 14 - 16, 2006, vol. 266. ACM, New York (2006)
7. Cai, Y., Miao, C., Tan, A., Shen, Z.: Fuzzy cognitive goal net for interactive storytelling plot design. In: Proceedings of the 2006 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2006), Hollywood, California, June 14 - 16, 2006, vol. 266. ACM, New York (2006)
8. Shen, Z.: Goal-oriented Modeling for Intelligent. Agents and their Applications, PhD Thesis, Nanyang. Technological University (2005)
9. Ciarlini, A.E., Pozzer, C.T., Furtado, A.L., Feijó, B.: A logic-based tool for interactive generation and dramatization of stories. In: Proceedings of the 2005 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2005), Valencia, Spain, June 15 - 17, 2005, vol. 265. ACM, New York (2005)
10. Barros, L.M., Musse, S.R.: Introducing narrative principles into planning-based interactive storytelling. In: Proceedings of the 2005 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2005), Valencia, Spain, June 15 - 17, 2005. ACM, New York (2005)
11. Crawford, C.: Chris Crawford on Interactive Storytelling. New Riders, Berkeley (2004)
12. Chaplin, D.J., El Rhalibi, A.: IPD for emotional NPC societies in games. In: Proceedings of the 2004 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2004), Singapore, June 03 -05, 2005, vol. 74. ACM, New York (2004)
13. Lankoski, P., Björk, S.: Gameplay design patterns for social networks and conflicts. In: Proceedings of the Fifth International Game Design and Technology Workshop, Liverpool, UK (November 2007)
14. Foundations of Interactive Storytelling [accessed February 2008], <http://www.igda.org/writing/InteractiveStorytelling.htm>
15. Crawford, C.: Storytron – Interactive Storytelling [accessed February 2008], <http://www.storytron.com/index.html>
16. Moreno-Ger, P., Blesius, C., Currier, P., Sierra, J., Fernández-Manjón, B.: Rapid development for game-like interactive simulations for learning clinical procedures. In: Proceedings of the Fifth International Game Design and Technology Workshop, Liverpool, UK (November 2007)
17. Mansilla, W., Jung, B.: Emotion and Acousmètre for Suspense in an Interactive Virtual Storytelling Environment. In: Proceedings of the Third International Game Design and Technology Workshop, Liverpool, UK (November 2005)
18. Rhalibi, A.E., Bendiab, A.T.: Harnessing Agent Based Games Research for Emerging Behaviour Analysis. In: Second GSFC/IEEE Workshop on Radical Agent Concepts (WRAC) Sponsored by IEEE CS Technical Committee on Complexity in Computing Corporate Sponsor – IBM, September 20-22, 2005. NASA Goddard Space Flight Center Visitor’s Center, Greenbelt (2005)
19. El Rhalibi, A., Baker, N., Merabti, M.: Emotional agent model and architecture for NPCs group control and interaction to facilitate leadership roles in computer entertainment. In: Proceedings of the 2005 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology (ACE 2005), Valencia, Spain, June 15 -17, 2005, vol. 265. ACM, New York (2005)

20. Sheldon, L.: *Character Development and Storytelling*. Thomson Course Technology, USA (2004)
21. Eladhari, M., Nieudorp, R., Fridenfalk, M.: The soundtrack of your mind: Mind music – adaptive audio for game characters. In: *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, Hollywood, California. ACM, New York (2006)
22. Bates, J.: The role of emotion in believable agents. *Communication of the ACM* 37(7), 122–125 (1994)
23. IGDA; *Foundations of Interactive Storytelling* [accessed March 2008], <http://www.igda.org/writing/InteractiveStorytelling.htm>
24. Bailie-De Byl, P.: *Programming believable characters for computer games*. Charles River Media, USA (2004)
25. Elliott, C.: *Multimedia communication with emotion-driven Believable-Agents*. In: *AAAI Technical Report for the Spring Symposium on Believable Agents*, pp. 16–20. AAAI, Stanford University (1994)
26. Hanneman, R.A., Riddle, M.: *Introduction to social network methods*. University of California, Riverside, CA (2005) [accessed March 2008], <http://faculty.ucr.edu/~hanneman/>
27. Hanseth, O., Monteiro, E.: *Understanding Information Infrastructure* [accessed March 2008], <http://heim.ifi.uio.no/~oleha/Publications/bok.html>
28. Puustinen, I., Pasanen, T.A.: *Creating realistic collaboration for Action Games*. In: *Proceedings of the Fourth International Game Design and Technology Workshop*, Liverpool, UK, November 15-16, 2006, School of Computing and Mathematical Sciences, Liverpool John Moores University, Liverpool, UK (2006)
29. Cohen, M., Riolo, R.L., Axelrod, R.: *The Emergence of Social Organization in the Prisoner’s Dilemma: How Context-Preservation and other Factors Promote Cooperation*. Working papers. 99-01-002. Santa Fe Institute (1999)
30. Pizzi, D., Cavazza, M.: *Affective storytelling based on character’s feelings*. In: *AAAI Fall Symposium on Intelligent Narrative Technologies*, Arlington, Virginia (November 2007)
31. Cavazza, M., Charles, F.: *Dialogue generation in character based interactive storytelling*. In: *AAAI First Annual Artificial Intelligence and Interactive Digital Entertainment Conference*, Marina del Rey, California, USA (2005)