# Directed Emergent Drama

Freyr Arinbjarnar

PhD

The University of York

Computer Science

September 2011

## Abstract

A fully interactive drama, where the player interacts with intelligent Non-Player Characters (NPCs), can revolutionise entertainment, gaming, education, and therapy. Creating such a genuinely interactive drama that is entertaining and gives players a sense of coherency as active participants in the unfolding drama has seen a substantial research effort. Authors have the power to shape dramatised stories for theatre or television at will. Conversely, authors ability to shape interactive drama is limited because the drama emerges from players' and NPCs actions during game-play, which significantly limits authoring control.

A coherent drama has a recognisable dramatic structure. One philosophy is to use planning algorithms and narrative structures to reduce required authoring. However, planning algorithms are intractable for the large state-spaces intrinsic to interactive dramas, and they have not reduced the authoring problem sufficiently.

A more straightforward and computationally feasible method is emergent interactive drama from players' and NPCs' actions. The main difficulty with this approach is maintaining a drama structure and theme, such as a mystery theme or a training scenario, that the player experiences while interacting with the game world. Therefore, it is necessary to impose some form of structure to guide or direct the unfolding drama.

The solution introduced in this thesis is to distribute the computation among autonomous actors that are guided by goals and drama structures which a centralised autonomous director agent distributes among the actors, which comprises the following four main elements:

a autonomous rational actor agents that know they are acting and can negotiate dialogues between them to remain realistic while simultaneously progressing the drama, without the player knowing,

b Bayesian network to model the actors reasoning, including beliefs about other actors' mental states

c an autonomous director agent uses "schemas", conceptual structures based on motifs, to guide the actors.

# Contents

# List of Tables

# List of Tables

# List of Figures

# List of Definitions



# List of Examples

**Acknowledgements**

To Tómas V. Albertsson who stood by me through it all from my very first Computer Science course, and has never lost faith in me.

To my dearest children who have patiently waited for dad to finally finish his big project and start to create exciting computer games.

To my mother who has also supported me with all her love as always.

Dr Christoforos Anagnostopoulos Head Of Research at Improbable (Enterprise) and Associate Professor at Imperial College who has provided exceptional support in the last months in the form of academic guidance, motivation and time off work to complete my thesis. Additionally his collaboration in constructing a test scenario to try out my engine with the Improbable Spatial OS, open source, technology. A scenario that is relevant to the enterprise departments current project goals.
Additionally the support of everyone at the Improbable's Enterprise department during these last months of finishing this PhD work.

Dr James Cussens, Senior Lecturer at the computer science department at York, for his steadfast support and encouragement along with expert academic advice on Bayesian inference.

Dr Daniel Kudenko, who started this PhD journey with me and supported my novel ideas that went against the current school of thought.

Additionally many members of the AIG (Artificial Intelligence Group) and the staff and students at the Computer Science department in York.

**Declaration**

This thesis is the original work of Freyr Arinbjarnar and describes research carried out at the University of York. Some of the material contained in this thesis has appeared in the following published book chapters and conference papers.

**Refereed book chapters**

**Actor bots.** Freyr Arinbjarnar and Daniel Kudenko. In *Believable bots*, Philip Hingston editors, Springer 2012.

**Refereed conference papers**

**Bayesian networks: Real-time applicable decision mechanisms for intelligent agents in interactive drama.** Freyr Arinbjarnar and Daniel Kudenko. In Proceedings of the *IEEE Conference on Computational Intelligence and Games (CIG2010)*, Copenhagen, Denmark, August 2010

**Duality of Actor and Character Goals in Virtual Drama.** Freyr Arinbjarnar and Daniel Kudenko. In Proceedings of the *9th International Conference on Intelligent Virtual Agents*, Amsterdam, Holland, September 2009

**Directed Emergent Drama vs. Pen & Paper Role-Playing Games.** Freyr Arinbjarnar and Daniel Kudenko. In Proceedings of the *AISB'09 Symposium: AI & Games*, Edinburgh,Scotland, April 2009

**A Critical View of Interactive Drama Systems.** Freyr Arinbjarnar, Heather Barber and Daniel Kudenko. In Proceedings of the *AISB'09 Symposium: AI & Games*, Edinburgh, Scotland, April 2009

**Schemas in Directed Emergent Drama.** Freyr Arinbjarnar and Daniel Kudenko. In Proceedings of the *First Joint International Conference on Interactive Digital Storytelling (ICIDS)*, Erfurt, Germany, November 2008

# 1

# Introduction

## 1.1 Motivation

In the last couple of decades, computer technology has evolved sufficiently to create MMOGs such as Neverwinter [Stu16], World of Warcraft [Ent04], Eve Online [Gam03], Runescape [Jag01]. In an MMOG, multiple players play simultaneously in the same game world and interact with each other, collaborate and even fight. Collectively the players create a virtual reality (VR), i.e. a virtual world where people socialise online. The Sims [Art19], VRChat [Inc17], IMVU [Rie04], Second Life [Lab03] (SL) are examples of virtual realities without a game element where players are motivated by:

- an alternative lifestyle,
- friendship,
- creating virtual goods that they give or sell to other players,
- running a full-time business; they exchange in-game currency for real-world currency on E-Bay.

Several single-player games, such as Vampyr [EI19], Assassin's Creed series [Ubi18], Elder Scroll series [LLC18] contain vast game worlds that a player can roam around in for months and still find new places to explore and new activities to occupy their time.

> "
>
> *What happens, in reality, is irreversible, while in play, it is possible to start again from scratch. Play is a simulation of reality*

> *that, far from making play a frivolous pastime, emphasises the immense importance of all play activity for the well-being and development of man.*
>
> *Children play in order to familiarise themselves with the behaviour patterns which they will have to use and experience in their life, in reality. Young animals play to learn to hunt, to flee, to orientate themselves. All play activity of this kind is inherently dramatic because it consists of mimesis, an imitation of real-life situations and behaviour patterns.*
>
> ”
>
> – [Ess76, p. 19-20]

Imagine an interactive emergent murder mystery where the player is the lead detective interviewing suspects and examining clues to discover the real perpetrator, a mystery that challenges the player's skills and intelligence. The player interacts with believable characters that respond fluently to dialogue and events.

Education can evolve to accommodate students' abilities. The students attend a VR classroom where they have unlimited access to teachers guidance, extra help and advanced material related to the student's actual performance, skill set, intelligence, motivation and abilities.

Significantly enhance therapy sessions by re-enacting patients' social challenges in a VR to improve therapists' assessments and patients' coping skills.

Envision real-world simulations where players' can interact with the NPCs that believably represent specific people or people similar to specific target groups from real-world events. Players can play significant roles in historical events, such as creating the human rights treaty or the Iraqi war and experience events from all sides.

Although some of these visions appear very distant, they all share a distinct common ground, which is the core of this proposal. They all call for interactive drama where a player actively participates as one of the characters in the drama's action experiencing each event first hand as it transpires and where the player can affect the scene such as to alter the course of future events.

**Definition 1.1.1 (Emergent interactive drama)**

> *For this thesis, we define that emergent interactive drama is a drama that emerges from one or more human actors' and NPCs' interactions, and comprises:*
>
> 1. ***Interactive drama:*** *A structured narrative which includes:*

- *Structuralism, e.g. Freytag's pyramid [Fre63], Proppian uniform structure and adherence to a known genre [Pro68].*
- *Drama theory e.g. Polti's situations [Pol21], and Improvisation.*
- *Drama: actions by one or more actors that play a role in front of one or more spectators.*

2. ***Players' agency:***
   - *Players' are either protagonists' or play a critical role in the drama.*
   - *Players' actions directly affect the course of the drama as expected within the confines of its genre.*
3. ***NPCs' believability,*** *Autonomous agents that demonstrate of their own volition:*
   - *Timely responses to players' actions.*
   - *Coherent response responses; respond in context to past and present events, their role, character, and emotional state.*
   - *Rational decisions; they choose actions according to Nash Equilibrium.*
   - *Theory of mind; they base their decisions on an understanding of what other characters think and how they may reply.*
   - *Have personality traits and emotions.*

Creating emergent interactive drama, as defined above, is a worthwhile challenge that can revolutionise entertainment, gaming, education, therapy and Defence stabilisation operations.

## 1.2 Background

Emergent interactive drama has two primary challenges; drama management and believable human behaviour representation.

**Drama management** depends on the aim of the drama, which includes:

1. Dramatising pre-authored stories where the player can either play the protagonist [YRB+04] or observe and influence the main characters [PCLC07].
2. A virtual training environment which ensures that the player encounters specific teaching moments [RS06c].
3. Generating a narrative structure from players' and NPCs' interactions [Szi03].

An alternative to drama management is unrestricted emergence with unpredictable results, which is unlikely to meet any of the above goals.

The main challenge in drama management is the need to author actions and dialogues in sufficient detail for the NPCs to respond correctly to players' interaction; coined as the authoring bottleneck [MS05b, RY06]. For example, it took three person-years to author Façade; which contains one scene, two characters and each drama takes approximately 25 minutes [MS05b].

Believable human behaviour representation is whether the NPCs exhibit behaviour similar to humans while maintaining the intended narrative structure.

As we defined in Section 1.1, we consider believable behaviour to be:

1. **Coherent**: The behaviour is consistent with previous behaviour and expectations based on character goals [Szi03, p. 6], and logically inferred [SGRR73] from its knowledge base.
2. **Rational**: As defined in Game Theory; that NPCs choose actions that optimise their utility given their expectation from a theory of mind reasoning.

The main challenges with creating believable NPCs is the complexity of the human mind and how to still manage the drama without compromising believability.

Solution: Our solution is the Directed Emergent Drama; to distribute the computation between autonomous and drama aware actor agents with a shallow director agent to create an emergent interactive drama.

## 1.3 Directed Emergent Drama (DED)

Fast-growing virtual realities and the apparent lack of development in playability in computer games in the last 15 years is the primary motivation for the DED and its contributions presented in this thesis. The DED has a director agent, actor agents and motif like structures called schemas that guide the emergence of an interactive drama. The drama emerges from the interactions of the actors and a human player. We summarise the DED below and give a detailed description in chapter 4.

**Schemas,** motif like structures similar to G. Polti's situations, that a director uses to inform the actors of where the drama should emerge to follow the drama genre and the structure laid out by an author. The schemas do not contain any specific instructions such as actions, sentences, gesture or behaviours. The director deploys several schemas in parallel. At any given time there are at least two or more schemas deployed, see detailed description in Section 4.3.

**Director,** shallow agent that matches schemas to actor agents to guide the emergent interactive drama. The director deploys schemas as appropriate for the intended drama genre. The director does not follow a pre-authored script similar to a film script. Instead, the director guides the actors using goals that are pre-assigned by an author, see detailed description in 4.4.

**Actors,** the drama's prima motor, interact with the player and improvise an emergent interactive drama. The actors are responsible for the drama and will overrule the director's guidance if it interferes with their performance. The actors are aware that they are actors in a drama and need to ensure that the drama conforms to the expected genre, see detailed description in Section 4.5.

**Player,** a human player that is simultaneously a spectator and an actor in the drama, i.e. the protagonist, see detailed description in Section 4.6.

## 1.4 Hypothesis

> That: *distributing drama management among rational, autonomous agents, using motif like structures called schemas, applying BNs as the core decision mechanism for the agents, giving the agents drama awareness and making them responsible for the drama emergence, will allow for real-time application of emergent interactive dramas that include several characters, behaviours and scenes, and run for more than half an hour*

We now define the terms in the hypothesis as we use them in this thesis:

"...distributing drama management among rational, autonomous agents..." - (novelty) - Decentralising, the director of the interactive drama systems with autonomous actors responsible for managing the drama, i.e. facilitating the expected emergent interactive drama.

"...using motif like structures called schemas..." - (novelty) - We use motif like structures similar to G. Polti's situations [Pol21], that are non-sequential and non-linear; they are an abstract depiction of a current situation. The schemas have goals that guide the actors in developing the drama, and goals that pertain to their character and cause conflict and tension. In this manner, the schemas generate a situation - evaluated by N. Szilas & Co computational models [SER16].

"...applying causal structured BNs as the core decision mechanism for the agents..." - (novelty) - We use rational agents capable of logical reasoning and theory of mind in an interactive drama system, building on our Rational Dialogue Engine (RDE) [Ari07a, Ari07b] - (evaluation) - Requires:

1. that players find them coherent
2. that recorded responses are rational BNs.

"...giving the agents' drama awareness and making them responsible for the drama emergence..." - (novelty) - Apply duality in the actor's decision mechanism, allowing the actors to be aware that they need to choose actions that are believable and aid the drama's progression—enabling the actors to realise if there is a conflict between believability and the drama requirements and ask other actors for help when needed.

"...will allow for real-time application..." - (evaluation) - Requires:

1. that players find them timely
2. that recorded reaction time is $<$ one second.

“...of emergent interactive dramas...” - (evaluation) - A drama emerges from players' and actors' interaction. Evaluated by observing the drama structure primarily from logs and by user testing.

“...that include several characters, behaviours and scenes - (evaluation) - The drama content is primarily observed from logs.

“...and run for extended periods...” - (evaluation) - the player can engage in the drama for at least 30 minutes; similar to a television drama.

We apply both computational and user evaluation that are grounded on the Compellingness criteria established by H. Barber [Bar08].

## 1.5 Thesis outline

Chapter 2. Background, we discuss relevant definitions and work from Narratology and how it relates to computer games. The DED builds on existing technologies and former work which we describe in some detail; starting with existing technology, followed by our former work.

Chapter 3. Related Work, We review related work and discuss existing challenges and how our work builds on lessons learnt.

Chapter 4. Directed Emergent Drama (DED), We describe the technical structure of the DED in detail; how the director, schemas, actors and player function within the emergent drama. We use the latest implementation of the DED in a leading MMOG toolset; SpatialOS, as a running example to demonstrate our solution.

Followed by a comparison of DED to related systems.

Chapter 5. Evaluation, We describe several DED applications, discuss their evaluation, and summarise the results.

Chapter 6. Conclusion, We summarise and discuss conclusions, limitations and future work.

# 2

# Background

## 2.1 Introduction

In this Chapter, we introduce and discuss several key definitions, technology and previous work for the DED. Therefore, We refer back to this Chapter throughout the thesis.

2.2 **Narratology**. We define narrative and drama, discuss their structure and how they relate to games. We end the section on a summary of the definitions and how they are relevant for the thesis.

2.3 **Computation.** We apply two core technologies in the DED:

  2.3.1 **Bayesian Network (BN)**; causally structured stochastic networks.

  2.3.2 **Game Theory (GT)**; rational and coherent reasoning when combined with BNs.

  A basic understanding of the underlying technology is imperative to understand our contribution.

2.4 **Former work**. The DED extends on our former published work:

  2.4.1 Bachelor of Science (BSc) project: Dynamic Plot Generating Engine (DPGE) [Ari06, Ari08].

  2.4.2 Master of Science (MSc) thesis: Rational Dialogue Engine (RDE) [Ari07a, Ari07b].

2.5 **Summary**. We finish the Chapter on a summary of its contents and how it leads towards our current work.

## 2.2 Narratology

We start by defining commonly used terms and how we understand them in the context of our work. A brief introduction to structuralism, and how we have established our definitions on the work of structuralists including: G. Freytag [Fre63] in Section 2.2.1.1 , G. Polti [Pol21] in Section 2.2.1.2, V. Propp [Pro68] in Section 2.2.1.3.

Additionally S. S. Van Dine [Van92], M. Esslin [Ess76], R. A. Fremann [Fre92], R. A. Knox [Kno92], G. Prince [Pri03], T. Todorov [Tod77] on the structure of the murder mystery, see Section 2.2.1.4. We define drama and the difference between drama and narrative, in Section 2.2.2, followed by a discussion about modern computer games and how they relate to drama and narrative in Section 2.2.3. We describe the structure of the murder mystery, see Section 2.2.1.4, which we repeatedly use in our experiments.

### 2.2.1 Narrative

The word narrative derives from the Latin verb narrare, which means "to recount", and, in everyday usage, it is a story or a novel written or told by a storyteller or a narrator. More specifically, a narrative presents a sequence of events which do not need to be causally connected [Pri03].

While a narrative tends to account past events rather than present events it is nevertheless a broad term that includes more than stories and novels according to the Oxford English Dictionary:

> "
>
> 1. *A spoken or written account of connected events; a story.*
>    - 1.1 *The narrated part of a literary work, as distinct from dialogue.*
>    - 1.2 *The practice or art of telling stories.*
>    - 1.3 *A representation of a particular situation or process in such a way as to reflect or conform to an overarching set of aims or values.*
>
> "
>
> – [Dic12]

Therefore, a narrative can include gameplay which has "an overarching set of aims or values" [Dic12]

Some narratives do not have an explicit causal sequence of events, for example, Ulysses [Joy22] where the narrator Leopold Bloom describes a series of mostly unconnected events that he witnesses while walking through Dublin on an ordinary day.

However, unlike Leopold, we have an intrinsic urge to reason about the world around us in narrative terms. To this end, we often imagine a narrative to explain current events, in the absence of actual action. For example, when looking at a static picture of a stranger, we contemplate several narrative aspects, including:

- who they are,
- what they are thinking,
- what they are feeling,
- What they are doing,
- what they intend to do,
- where they came from, and
- where they are going.

We aim to develop algorithms that generate narratives from users' interactions with events and characters in virtual realities. Algorithms need structure to facilitate meaningful emergence. Structuralism is a field within Narratology dedicated to identifying intrinsic structures within the literature.

We give a brief introduction to relevant structuralists and their work:

1. G. Freytag [Fre63] in Section 2.2.1.1
2. G. Polti [Pol21] in Section 2.2.1.2
3. V. Propp [Pro68] in Section 2.2.1.3
4. Structuralistic approach to analysing the murder mystery.

#### 2.2.1.1 Freytag's pyramid

In 1863, a German playwright and novelist Gustav Freytag wrote Die Technik des Dramas, a definitive study of the dramatic arc in five acts as follows:

1. Exposition (originally called introduction)
2. Rising action (rise)
3. Climax
4. Falling action (return or fall)
5. Denouncement, resolution, revelation, or catastrophe

The pyramid is an excellent baseline to understand the general flow of a drama and the need to build suspense via complex interactions of the characters. It is unnecessary to stick strictly to G. Freytag's five acts; he would likely have modified his structure if he had been around when television and movies started to be popular. After all, there are multiple acts in a TV series, neither is it always five acts within an episode. In our modern medium, we need to adjust the structuralist theories to the current conditions that the authors could not foresee.

#### 2.2.1.2 Polti's 36 situations

According to G. Polti, there are 36 situations [Pol21] that represent every dramatic situation known in drama, for example, the below list with the numbers from Polti's list:

- (23) Necessity of sacrificing loved ones; a hero wrongs a beloved victim because of the necessity for their Sacrifice.
- (24) Rivalry of superior vs inferior; a superior beat an inferior rival and wins the prise
- (27) Discovering the dishonour of a loved one; the discoverer discovers the wrongdoings committed by the Guilty One.
- (30) Ambition; the ambitious person is opposed by an adversary when seeking the coveted thing.
- (33) Erroneous judgment; the mistaken one falls victim to the cause or the author of the mistake and passes judgment against the victim of the mistake when it should be passed against the guilty one instead.
- (34) Remorse; the culprit wrongs the victim or commits the sin, and is at odds with the interrogator who seeks to understand the situation.
- (35) Recovery of a lost one; the seeker finds the one found.
- (36) Loss of loved ones; a kinsman sees their kin slain in front of them.

These situations are motifs of the conflict between characters producing the drama.

#### 2.2.1.3 Propp's Morphology

Vladimir Propp [Pro68] a Russian structuralist analysed a subset of Russian folktales, classified by Aarne index 300 to 749 [Aar11].

Arne index is a collection of folktales indexed to enable researchers access to preserved copies of original folktales. Propp selected a specific set of the folktales from the index, i.e. tales 300 to 749. These tales were so similar that he could reduce the number of folktales from 449 to a set of 100 distinct tales.

Propp then further analysed these 100 folktales and found that, although names and attributes of the characters change, they all adhered to the same structure which he captured in the following four rules [Pro68]:

1. Functions of characters serve as stable, constant elements in a tale. They constitute the fundamental components of a tale.
2. The number of functions known to a fairy tale is limited.
3. The sequence of functions is always identical.

4. All fairy tales are of one type regarding their structure.

Propp proved that this is true for the tales in Aarne index 300 to 749 by showing that it holds for every single one of those tales. Propp named and numbered the functions that make up these specific folktales. The result is called Propp's morphology of the Russian folktale. There could exist Russian folktales that are not of this structure simply because they were not in the Aarne index.

Propp further theorised, in line with other structuralists, that there exists a uniform structure for stories, i.e. that all stories, of any genre, will conform to a single distinct structure. Due to the theories of the structuralist of generalised and uniform structure in all things, their work is of significant interest to Computer Scientists when creating algorithms that rely on structure and predictable patterns.

#### 2.2.1.4 The murder mystery

A typical murder mystery is in three acts, a prologue, a sizeable middle part and an epilogue [Tod77].

**In act I,** the prologue, the drama introduces the characters and sets the scene; including any secret drawers, hidden compartments and any readily accessible stacks of poisons or lethal apparatus. Additionally, the detective discovers the body, traditionally in one of the last scenes after the drama introduces the victim.

**In act II,** the detective interviews all the suspects and observes all relevant clues to deduct who is the murderer. All clues need to be accessible to the detective by the end of act II to carry out their role. The game should reveal to the detective all suspects' motives, means, and opportunity through either dialogue or physical evidence.

**In act III,** the epilogue, the murderer is revealed by showing that only the murderer had a motive, means and opportunity. We have identified specific primary goals that should be satisfied to progress the drama, from act I to act II and act II to act III.

The drama has five primary sections that form the dramatic arc: exposition, complication, climax, fall, and closure [Pri03]. In the murder mystery, the exposition or prologue is the first act. It serves to introduce characters and scenery and has at its end a murder that serves as the inciting moment of the drama. The second act contains the plot's complication, where many facts about the characters and their relationship to the murder surface. Events first occur in a seemingly disparate way, but as act II progresses, the facts become more evident, and gradually reveal the events' sequence before the murder. The final act sees the climax of discovering the murderer and the closure of proving that no other could be the murderer.

The murder mystery genre is a game of *spotting the piece that does not fit* [Kno92, Tod77]. A good mystery has all clues in the open easily observable but not visible. The detective needs to recognise their significance to the mystery. The clues appear ordinary and blend into the background; the detective needs to discover them via an investigation.

It is a game of pretence; characters appear manipulative and devious without the viewer recognising it because they do not see the real motive for the character's behaviour. Therefore, the viewer frequently sees intelligent devious characters as carefree, superficial, and somewhat silly [Van92, Ess76].

There is a pretence of normality in the murder mystery surrounding the bizarre incident of having someone brutally murdered in and ordinary everyday scene. The drama depicts the suspects as good standing citizens that we would typically not consider capable of murder [Fre92, Van92, Ess76].

When we talk about revealing characteristics of characters, then we refer to this pretence. Namely, characteristics are revealed but not in direct connection to the real motive behind the characters actions. The detective will need to connect motives with the actions and characteristics of the suspects. The same applies for revealing clues; The clues should be observable, but the detective needs to understand how they can be a piece in the puzzle and put the pieces together to make a whole picture.

### 2.2.2 Drama

In the Oxford English Dictionary drama is defined as follows:

> *" A composition in prose or verse, adapted to be acted on stage, in which a story is related by means of dialogue and action, and is represented, with accompanying gesture, costume and scenery, as in real life; a play. "*
>
> – [Dic12]

This definition is acceptable for drama proper, that is a drama for a theatre stage, but there are many other types of drama, and for those, as M. Esslin says, this is "downright incorrect" [Ess76, p. 9] for the following reasons:

1. "A composition in prose or verse" [Dic12]. According to M. Esslin: "seems to imply a text previously composed" [Ess76, p. 9] which would rule out any improvised drama.
2. "adapted to be acted on stage" [Dic12]. In M. Esslin's words: "what then about drama in television, radio or the cinema?"[Ess76, p. 9] if drama requires a stage, then none of these would constitute a drama.

3. “in which a story is related by means of dialogue and action" [Dic12]. If a dialogue is necessary, then mime cannot constitute drama and neither would many of the old black and white movies that had no sound. Neither is a physical action necessary since it rules out radio drama.
4. “and is represented, with accompanying gesture, costume and scenery" [Dic12]. M. Esslin says: “I have seen very good drama without costume or indeed scenery" [Ess76, p. 9] and neither costume nor scenery are in radio drama.
5. “as in real life" [Dic12]. M. Esslin says: “It seems to assume that all drama must be realistic drama" [Ess76, p. 9], *Waiting for Godot* [Bec53] and *The Merry Widow* [Leh05] are not like real life but are still dramas.

According to M. Esslin drama is a broad term that encompasses most of human behaviour, he says: “It is well-nigh impossible to draw the exact dividing line between when one kind of general activity stops and drama proper starts." [Ess76, p. 10]. Therefore, drama includes activities such as

- ritual,
- make-believe play,
- tribal dances,
- religious services,
- great state occasions,
- gladiatorial contests,
- public executions, and
- spectator sports,

However, according to M. Esslin's they are: “not proper drama such as we see in theatre with the full dramatic structure"[Ess76, p. 10].

For example, a soccer game matches the Oxford English definition of drama. A soccer game has an illuminated and clearly defined stage. The game has spectator that applaud and cheer and boo if the game goes badly, and it has a clear beginning and end, with a rise and fall in suspense. There are clearly defined gestures, dialogues, actions, scenery, and costumes in a soccer game.

Drama is primarily an action, “in Greek the word *drama* simply means *action*. Drama is a mimetic action, action in imitation or representation of human behaviour"[Ess76, p. 14]. Since drama is at its heart action, then actors are an essential part of the drama, “there is no drama without actors whether they are present in flesh and blood, or projected shadows upon a screen, or puppets"[Ess76, p. 11]. In order for there to be action, there must be an actor to act, we only need one actor; "even the internal monologue, where we follow the thoughts of a single character is drama and is frequently acted, particularly on radio"[Ess76, p. 19].

Drama "puts the spectator into the same situation as the character to whom the words are addressed." [Ess76, p. 18] Effectively for our purpose drama is; action by one or more actors that play a role in front of one or more spectators. In the words of M. Esslin:

> *" The author and the performers are only one half of the total process: the other half is the audience and its reaction. Without an audience there is no drama. "*
>
> – [Ess76, p. 23]

We start our lives learning through play, action and interaction. Drama in our digitalised world, a structured play that allows us to learn and grow in a manner that is instinctive from earliest childhood is our new frontier in entertainment, training and therapy. In M. Esslin's words:

> *" Through the mass media drama has become one of the most powerful means of communication between human beings, far more powerful than the merely printed word which was the basis of the Gutenberg revolution.*
>
> *That is why a knowledge of the nature of drama, an understanding of its fundamental principles and techniques and an ability to think and talk about it critically has become very necessary indeed in out world and that does not only apply to such great works of the human spirit as the plays of Sophocles or Shakespeare, but also to the television situation comedy or, indeed, to that briefest of dramatic forms, the television or radio commercial. "*
>
> – [Ess76, p. 12]

#### 2.2.2.1 Improvisation

Improvisation is free form drama that does not have the rigid structure of drama proper; there is no script or pre-authoring, no director that micromanages every move as K. Johnstone says:

> *" I'd argue that a director should never demonstrate anything to an actor, that a director should allow the the actor to make his own discoveries, that the actor should think he'd done all the work himself. I objected to the idea that the director should work out the moves before the production started. I said that if an actor forgot a move that had been decided on, then the move was probably wrong. Later I argued that moves weren't important, with that*

> *only a couple of actors on a stage, why did it matter where they moved anyway? I explained that **Hamlet** in Russian can be just as impressive, so were the words really of first importance? I said that the set was no more important than the apparatus in the circus. "*
>
> – (FA: emphasis is in italics in the original text) [Jho79, p. 24]

While improvisation does not always lead to success, it often outshines pre-authored text, as K. Johnstone says:"We learned that things invented on the spur of the moment could be as good or better than the texts we laboured over"[Jho79, p. 26]. It is not necessary to get the authored text correctly for drama is action and not a literary art as K. Johnstone says:

> *" Many writers of great talent have failed to write successful plays (Blake, Keats Tennyson, among others) because of a failure to understand that drama is not primarily a literary art. Shakespeare is a great writer even in translation; a great production is great even if you don't speak the language. A great play is a virtuoso display of status transactions - **Waiting fo Godot**, for example. "*
>
> – (FA: emphasis is in italics in the original text) [Jho79, p. 72]

Improvisation is spontaneous, an interaction between actors that collaborate to progress the drama, as in this example given by K. Johnstone:

> *" If an improviser is stuck on an idea, he shouldn't search for one, he should trigger his partner's ability to give 'unthought' answers. If someone starts a scene by saying 'What are you doing here?' then his partner can instantly say, without thinking, 'I just came down to get the milk, Sir.'*
> *'Didn't I tell you what I'd do if I caught you again?'*
> *'Oh Sir, don't put me in the refrigerator, Sir.'*
> *If you don't know what to do in a scene, just say something like, 'Oh my God! What's that?'*
> *This immediately jerks images into your partner's mind 'Mother!' he says, or 'That dog's messed the floor again!', or 'A secret staircase!' or whatever. "*
>
> – [Jho79, p. 82]

In this manner, improv actors give each other ideas and seeds to create more drama. They collaborate and play together, passing over to each other suggestions on how the plot can develop while they simultaneously remain in character (IC). Which is called not to block the other actors' suggestions as K. Johnstone describes it:

> *" There are people who prefer to say 'Yes', and there are people who prefer to say 'No'. Those who say 'Yes' are rewarded by the adventures they have, and those who say 'No' are rewarded by the safety they attain. There are far more 'No' sayers around then 'Yes' sayers, but you can train one type to behave like the other.*
> *'Your name Smith?'*
> *'No.'*
> *'Oh ... are you Brown, then?'*
> *'Sorry.'*
> *'Well, have you seen either of them?'*
> *'I'm afraid not.*
> *Whatever the questioner had in mind has now been demolished and he feels fed up. The actors are in total conflict.*
> *Had the answer been 'Yes', then the feeling would have been completely different.*
> *'Your name Smith?'*
> *'Yes.'*
> *'You're the one who's been mucking about with my wife then?'*
> *'very probably.'*
> *'Take that, you swine.*
> *'Augh!'*
> *Fred Karno understood this. When he interviewed aspiring actors he'd poke his pen into an empty inkwell and pretend to flick ink at them. If they mimed being hit in the eye, or whatever, he'd engage them. If they looked baffled, and 'blocked' him, then he wouldn't. "*
>
> – [Jho79, p. 92]

To block a suggestion can stop the plot from developing in a specific direction while accepting an offer opens new paths, new complication, and increased suspension. For this reason, the actors benefit from accepting an offer and use it to develop the plot further as in K. Johnston's words:

> *" Bad improvisers block action, often with a high degree of skill.*
> *Good improvisers develop action:*
> *'Sit down, Smith.'*
> *'Thank you, Sir.'*
> *'It's about the wife, Smith.'*
> *'She told you about it has she, Sir?'*
> *'Yes, yes, she's made a clean breast of it.'*
> *Neither actor is quite sure what the scene is about but he's willing to play along, and see what emerges. "*
>
> – [Jho79, p. 95]

Furthermore, to accept does not mean to mindlessly do what another actor asks but to use the request as a means to build the plot. Similarly to block does not mean to say no or to disobey the other character; instead, it means to refuse to acknowledge and accept the part offered as K. Johnston says:

> *" I call anything an actor does an 'offer'. Each offer can either be accepted, or blocked. If you yawn, your partner can yawn too, and therefore* ***accept*** *your offer.*
> *A block is anything that prevents the action form developing, or that wipes out your partner's premise. If it develops the action it isn't a block. For example:*
> *"Your name Smith?"*
> *"What if it is, you horrible little man!"*
> *This is not a block, even though the answer is antagonistic. Again:*
> *"I've had enough of your incompetence, Perkins! Please leave."*
> *"No, Sir!"*
> *This isn't a block either. The second speaker has accepted that he's a servant, and he accepts the situation, one of annoyance between himself and his employer.*
> *If a scene were to start with someone saying "Unhand me, Sir Jasper, let me go", and her partner said "all right, do what you like, then". this is probably a block. It would get a laugh but it would create bad feeling. "*
>
> – (FA: emphasis is in italics in the original text) [Jho79, p. 97]

A block or an accept action is, therefore, not necessarily visible to the audience unless they understand the structure of improv, and even then; skilful actors mask the interaction with a masterful play of their characters.

#### 2.2.2.2 Interactive Drama

In Section 2.2.2 we argued that drama is, in essence, an action, acted by one or more actors in front of an audience [Ess76].

Interactive drama is a drama where one or more spectators from the audience play a part in the drama as if they were one of the actors, even if their role is nothing more than stating their name, posing or holding something at the request of one of the actors. Although a spectator has taken on a role in the drama, they remain a spectator of the drama because they are only improvising without a script. They do not know how the drama will play out unless they have seen it before. Therefore, the participating audience is still spectating the drama as it unfolds, which means that it remains a drama; it retains all the suspense and tension that drama can have.

We must not to be too rigid in our definitions because that will only restrict the growing field that interactive drama is, as Esslin expertly puts it:

> *" Definitions–and thinking about definitions–are valuable and essential, but they must never be made into absolutes; if they are, they become obstacles to the organic development of new forms, experiment and invention. "*
>
> – [Ess76, p. 11]

Interactive drama is a new form of media, and we have, to a large extent, made the same mistakes as in the first films when they filmed theatre productions without adapting them to the altered viewpoint of the camera lens, which resulted in early films being over-acted, i.e. the actors grossly overdid displays of emotions and character because they needed them to be visual to a theatre audience. The camera lens, on the other hand, is close to the actor's face and catches every tiny facial expression which forced the actors to adopt an entirely new way of displaying emotions that would never work on a stage. In the same manner, we should seize this chance to evolve drama into this new media, creating new genres in drama, education, therapy, entertainment and news coverage.

In essence, an interactive drama is an improv designed to engage the audience in the drama.

### 2.2.3 Game

> *" …the vicarious play activity (which drama is for adults)… "*
>
> – [Ess76, p. 21]

Game worlds become increasingly advanced in graphics and sound, and players can spend months both in single-player and online-mode exploring, taking on quests and interacting with NPC (NPCs). There are mainly three computer game genres that are of interest here:

1. FPSs games are centred on using a gun or any projectile weapon and with it killing a large number of enemies that are bent on killing the player. The games are thus very action-based, and the player needs good coordination and reflexes to master the challenges of the games. There is usually the main storyline that runs through the game and multiple levels that the player needs to complete to progress in the game.
2. Adventure games are quite the opposite of FPSs; it is rare that action or speed is required to complete the game. Which is why they are frequently called "point & click" games because for the most part all

the player does is to locate items with the mouse and combine items to progress the story or talk to NPCs in the game through drop-down dialogue boxes. They are very puzzle-oriented, and the player frequently needs to solve several puzzles to complete the game. These games tend to be story centred and based on a film or a novel.

3. Role-Playing Games (RPGs) are a combination of FPS and adventure games and Pen & Paper Role-Playing Games (PP-RPGs) which is the "real-life version" of Computer-Based Role-Playing Games (CB-RPGs). In an RPG the player takes on a role in a rather rich virtual world and through a series of quests will save this world from impending doom or resolve some significant in-game issue. The player will gain experience through solving quests and killing monsters and increase in power (level up) as the game progresses. These quests and levelling up combine elements from FPSs and Adventure games. Players' require skill to kill the monsters encountered, and the player also needs to pick up objects and solve puzzles. RPGs have a significant story element; there is the main plot that needs to be played through to complete the game and multiple short stories in the quests that the player takes on. In PP-RPGs, a group of players are needed to play the game using pens and papers to keep track of their character actions and experience gain. There is also the need for a Game Master (GM) who tells the story of the game, plays the NPCs and in general, takes care of managing the gameplay.

At the dawn of CB-RPGs, there where high hopes for the extension of PP-RPGs into computer games. There were visions of vast worlds with endless possibilities and fruitful interaction with NPCs, possibly meeting favourite heroes from favourite fantasy series. The player would not be limited by the imagination and storytelling skills of the GM or by how successfully a good group of players can gather. The players could adopt any role they desired, shape the character to their likings and play through virtual reality, receiving a unique play experience for each character that they play. Included in the unique play experience would be a clear sense of a story and purpose for the characters actions just like a quality GM can create in PP-RPGs.

The primary motivation for this vision is that it can be challenging to find a qualified GM that offers sessions in the worlds and playing-style of individual player's likings. It can also be challenging to find a good group that fits the player's playing style. Players are primarily seeking is an RPG tailored to their tastes, competence and play style in the CB-RPGs.

CB-RPGs often fall short of offering the interactivity of PP-RPGs, as G. Gygax says:

> *" Pen-and-paper role-playing is live theatre and computer games are television. "*

– As quoted in [Sch06]

Pre-authoring is named by some as the primary reason, in the words of B. Coffin:

> “ *Sometimes G.M.s will write out an adventure's entire plot, soup to nuts. This is a dangerous thing, because it inherently contradicts one of the fundamentals of role-playing: the players should be allowed to determine the pace and direction of the adventure.* ”
>
> – [Cof01]

Similarly, many CB-RPGs railroad the player to choose specific actions in order to complete quests and progress in the game's main story, and the NPCs' responses are pre-authored, making them very repetitive. The results are rich game worlds that lack replayability due to repetitive storylines and limited NPC interactions.

RPG Neverwinter Nights [Bio02] uses the alignment of the character to determine the available paths through the game. Each player creates a character and determines the initial alignment. The players interact with NPCs and choose between good and evil actions. When players choose evil actions, their alignment shifts to evil, and if they choose the right action, their alignment shifts towards good. Quests that are accepted and completed are also used to adjust players' alignment. Good deeds adjust the players' alignment to good and evil deeds towards evil. The alignment affects the player's progress in the game by for instance hindering players playing druids in getting full access to their powers and the druid community if the player has not been careful to keep the character alignment as good. It will also affect what sub-quests are available, especially when it comes to quests that are for the advancement of a specific role and need a specific alignment. Still, it does not change the main story or the main story plot; the player will need to finish specific main quests in order to advance and complete the main story. Nothing the player does will affect the pre-authored plots of the game.

Fallout 3 [LLC02] has brought this idea even further; the player's actions significantly affect the character's reputation and opportunities. The availability of quests is dependant on player's actions; how much they explore the world, what they say in conversation, and what skills they pick when levelling up. Still the same can be said for Fallout 3 as with Neverwinter nights: nothing the player does will change the pre-authored plots or the main storyline. The player is always playing along one branch of a multilinear story that mostly follows the same fundamental plot-points each time. Which means that players will not get a novel story experience when replaying the game, which makes it very repetitive and lacking replayability.

In multilinear games, the players can save and reload if they say something to an NPC that has negative results. Equally, if they fail in a battle or a quest, they can reload an earlier save. The game adapts to the players and the players adapt their playstyle to the game. Which is very natural behaviour on for the player, and should be accommodated. Furthermore, some players will systematically try every action and explore every area and put their acquired information on the web where other players can benefit from it if they choose.

Games are intrinsically similar to an interactive drama which at its core is “a play”.

In the words of Gary Gygax:

> *“ Pen-and-paper role-playing is live theatre, and computer games are television. ”*
>
> – [Sch06]

As we have discussed drama is an integral part of RPGs, perhaps surprisingly drama is also becoming more and more integrated into First-Person Shooter (FPS) games.

FPS games are not very replayable in their basic single-player form. The reason for this is that after the first playthrough then playing it again, reveals nothing new and there is no suspense. There are the same bosses, and there are no significant changes between each gameplay and no new challenges. Multi-player FPS is, on the other hand, highly replayable because then the player is always up against a new challenge. In recent years single-player FPS games have evolved towards having the opponents, the NPCs, more intelligent to provide a more significant challenge for the player. They will simulate the normal behaviour of people in similar situations, including going to sleep, eating and in general going about their daily lives. One of the most recent and hugely popular games in these series is Assassin’s Creed II [Ubi09] where the player is an assassin that is assigned several assassination jobs. In Assassin’s creed as in many other recent FPS, there is a much greater emphasis on the story and character development than before. The player gets an in-depth character description of the Assassin and why he is in this job. The story progresses gradually through the game with seamless cut scenes. The player is playing the protagonist, and as the story progresses, the protagonist faces many moral dilemmas and challenging revelations that dramatically change his perspectives and shape his character. It is a common theme of other recent games such as Fallout 3 [LLC02] and Heavy rain [Ent09]. Heavy rain carries this character development even further and allows the player to make decisions based on the characters’ emotions and thoughts, forcing the player to make challenging moral decisions. There is an evident trend towards making the NPCs more intelligent and more autonomous in order to give the player more fulfilling game experience. An excellent example of this is STALKER [Wor09], there the NPCs actively simulate the behaviours of soldiers or mercenaries that try their best to kill the player in heavily polluted Chernobyl. Their behaviour is very believable, and they provide challenging gameplay, they do not merely continuously walk the same circle. In this respect, there is an apparent disparity between the increased playability of FPS which has every potential of providing rich replayability, as becomes clear when these games offer multi-player mode and the still pre-scripted unchanging storyline that the player is forced to follow.

### 2.2.4 Summary

We discussed how the combination of narrative and drama theory underlies our work for interactive drama and provides our algorithms' structure. Especially the Structuralists' work that we use to evaluate whether the emergent interactive dramas have the expected structure for their genre:

1. Freytag's pyramid.
2. Polti's 36 situations.
3. Propp's Morphology.

Moreover, we introduced the template that we use to evaluate whether the murder mystery emergent interactive dramas conform to their genre.

## 2.3 Computation

We introduce DED's core technologies, starting with Bayesian Network (BN) in Section 2.3.1, which we use as the basis for actor agents knowledge base and coherent reasoning, followed by Game Theory (GT), in Section 2.3.2, which we use for the actors rational reasoning algorithms that are a powerful tool to create believable characteristic behaviour. BNs' and GT's powers are then combined in the MAIDs, in Section 2.3.2.4, creating an exact and accessible means of developing robust and coherent reasoning.

### 2.3.1 Belief Networks

Belief Networks are probabilistic graphs comprised of vertices and directed arcs that indicate conditional dependency. Their purpose is to represent a belief, rather than certainty, about some state or event. For example, to form a belief about what has caused the lawn to get wet; was it the sprinkler or did it rain.

Since Belief Networks use probability distributions, rather than absolute boolean logic, they are especially suitable for incorporating any uncertainty which we commonly find in reasoning about the world. Moreover, they are well suited for challenging psychological models of traits and emotions which are very important for believability [BB00, BH09, GD07, HP99, LWHS07, TG01] when creating characters for drama.

A BN is a type of Belief Network in the form of a directed acyclic graph (DAG); comprised of vertices and arcs that indicate conditional dependency [Jen01].

We use GeNIe an academic modelling environment [DSL07a] for the graphical representation of BN models used in this thesis, and Smile an academic BN C++ library for calculations [DSL07b].

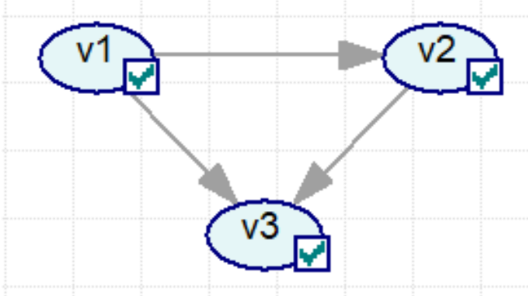
#### 2.3.1.1 The graph structure

The structure of a BN is as follows:

- A finite set of vertices $V(DAG) = v_1, ..., v_n$ where n is the number of vertices.
- A finite set of pairs of vertices called Arcs, $A(DAG) = \{(v_1, v_2), (v_1, v_3), (v_2, v_3)\}$
- Where the directed arcs are considered to start in the former vertex and end in the latter. For example, the arc $(v_1, v_2)$ starts in $v_1$ and ends in $v_2$. This is represented with an arrow leading from $v_1$ to $v_2$, in Figure 2.3.1.
- We call $v_1$ the parent and $v_2$ the child in the arc $(v_1, v_2)$.

- We call a vertex with no parent a root vertex. In our current example $v_1$ is the only root vertex.
- There cannot be any path in the graph, if travelling in the direction of the arrow from one vertex to another consecutively, that returns to a vertex that has been previously visited along that path.
- There is only one arc between each pair of vertices.

**Figure 2.3.1** Directed arcs in a BN, displayed in the GeNIe UI [DSL07a]

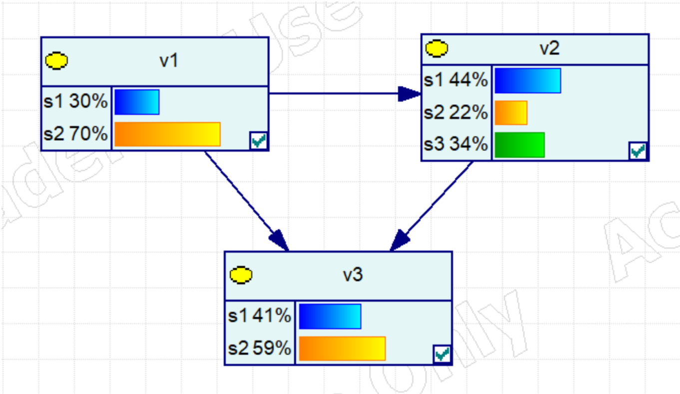



#### 2.3.1.2 The vertices structure

Each vertex has a set of states that can be either continuous or discrete. For this thesis, we only require discrete states and will, therefore, not cover continuous states. In the examples below, we have created simple examples with made up values for the introduction of how to evaluate BNs. The states of a vertex are:

- A finite countable set of states for each vertex $S(v) = s_1, ..., s_n$ where n is the number of states for that vertex.
- There are at least 2 states per vertex.
- Each set of states has a marginal probability distribution that in total equals 1 or 100% for any given vertex. For example, if $v_1s_1 = 0.3$ and $v_1s_2 = 0.7$ then $0.3 + 0.7 = 1$ in Figure 2.3.2.
- If a state's marginal probability has the value of 1 it is considered to be true and if it has the value of 0 it is considered to be false.
- States for a vertex are mutually exclusive, if the marginal probability of $v_1s_1 = 1$ then $v_1s_2 = 0$.

**Figure 2.3.2** A simple BN with 3 vertices and 3 arcs, displayed in the GeNIe UI [DSL07a]




#### 2.3.1.3 Assigning conditional probability distribution

Each vertex requires a definition of each of its conditional probabilities in order to evaluate its marginal distribution. We define the conditional probability of a vertex as follows:

- A root vertex has only the assigned marginal distribution since it has no parents, see Table 2.3.1 for $v_1$,s values and its visual representation in Figure 2.3.2.
- For $v_1$ we say that the probability (P) of either of its state being true is $\mathrm{P}(v_1 s_1) = 0.3$ and $\mathrm{P}(v_1 s_2) = 0.7$.
- For vertices $v_2$ and $v_3$ we need to define their conditional probability distribution given each of their parent states, see Table 2.3.2 for $v_2$'s values and Table 2.3.3 for $v_3$'s.
- For vertex $v_2$ we say that the probability of $s_1$ being true given its parent $v_1 s_1$ being true is $\mathrm{P}(v_2 s_1 | v_1 s_1) = 0.3$ and similarly the $\mathrm{P}(v_2 s_3 | v_1 s_2)$ $= 0.4$, see Table 2.3.2.
- When there are more than one parent we need to define a conditional probability distribution given every permutation of the parents states.
- For vertex $v_3$ we say that the probability of $s_1$ being true given its parents $v_1 s_1 \cap v_2 s_1$ being true is $\mathrm{P}(v_3 s_1 | v_1 s_1 \cap v_2 s_1) = 0.1$ and similarly the $\mathrm{P}(v_3 s_2 | v_1 s_2 \cap v_2 s_3) = 0.6$, see Table 2.3.3.
- The vertices are also called chance vertices because they define the conditional probability distribution.

**Table 2.3.1** Marginal probability distribution for $v_1$.

| | |
|---|---|
| $s_1$ | 0.3 |
| $s_2$ | 0.7 |

**Table 2.3.2** Conditional probability distribution for $v_2$.

| $v_1$ | $s_1$ | $s_2$ |
|---|---|---|
| $s_1$ | 0.3 | 0.5 |
| $s_2$ | 0.5 | 0.1 |
| $s_3$ | 0.2 | 0.4 |

**Table 2.3.3** Conditional probability distribution for $v_3$.

| $v_1$ | $s_1$ | | | $s_2$ | | |
|---|---|---|---|---|---|---|
| $v_2$ | $s_1$ | $s_2$ | $s_3$ | $s_1$ | $s_2$ | $s_3$ |
| $s_1$ | 0.1 | 0.2 | 0.3 | 0.6 | 0.5 | 0.4 |
| $s_2$ | 0.9 | 0.8 | 0.7 | 0.4 | 0.5 | 0.6 |

**Definition 2.3.1 (Marginal probability [Jen01])**

$$P(A) = \sum_{i=1}^{I} P(A|x_i)P(x_i)$$

See Definition 2.3.1 where $P(A)$ is the probability of a given state, and $I$ is the number of permutations of all the parents' states and $X = x_1, ..., x_I$ is the set of all permutations of parent states.

We can now evaluate the marginal probability distribution of $v_2$ by aggregating the conditional probability of each state given the marginal distribution of its parent vertices, see Table 2.3.4

**Table 2.3.4** Marginal probability for $P(v_2s_2)$

| | | | | |
|---|---|---|---|---|
| $P(v_2s_2)$ | $=$ | $P(v_2s_2\|v_1s_1)P(v_1s_1)$ | $+$ | $P(v_2s_2\|v_1s_2)P(v_1s_2)$ |
| $P(v_2s_2)$ | $=$ | $0.5 * 0.3$ | $+$ | $0.1 * 0.7$ |
| $P(v_2s_2)$ | $=$ | 0.22 | | |

Given the marginal probability distribution of $v_1$ and $v_2$ we can now use it to similarly calculate the marginal distribution of $v_3$, see Table 2.3.5

**Definition 2.3.2 (Bayes' rule [Jen01])**

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

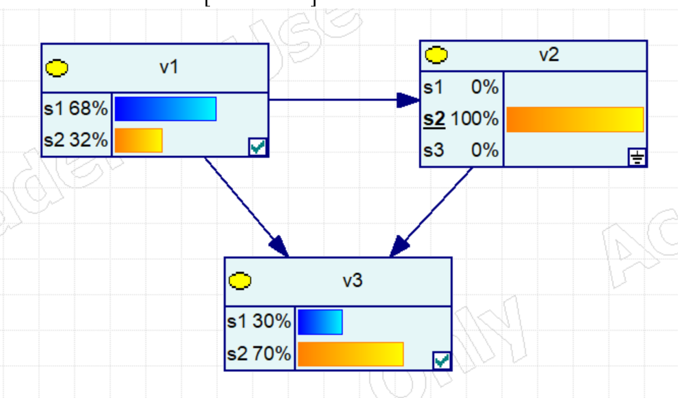
**Table 2.3.5** Marginal probability for $P(v_3s_1)$

| | | |
|---|---|---|
| $P(v_3s_1)$ | $=$ | $P(v_3s_1\|v_1s_1 \cap v_2s_1)P(v_1s_1)P(v_2s_1)$ |
| | $+$ | ... |
| | $+$ | $P(v_3s_1\|v_1s_2 \cap v_2s_3)P(v_1s_2)P(v_2s_3$ |
| $P(v_3s_1)$ | $=$ | $0.1 * 0.3 * 0.44$ |
| | $+$ | $0.2 * 0.3 * 0.22$ |
| | $+$ | $0.3 * 0.3 * 0.34$ |
| | $+$ | $0.6 * 0.7 * 0.44$ |
| | $+$ | $0.5 * 0.7 * 0.22$ |
| | $+$ | $0.4 * 0.7 * 0.34$ |
| $P(v_3s_1)$ | $=$ | $0.41$ |

BNs use Bayes' rule, see Definition 2.3.2, to evaluate the conditional probability distribution of vertices when the network is updated with evidence as follows:

- A vertex has evidence when we set one of its states marginal probability to 1.
- We call setting evidence to "instantiate the state".
- We also say that an instantiated state is a "true state".
- The instantiated state $v_2s_2$, in Figure 2.3.3, affects the conditional probability distribution of its parent and child vertices.

**Figure 2.3.3** A BN with an instantiated state, $v_2s_2$, displayed in the GeNIe UI [DSL07a]




We already know how to evaluate the child's marginal distribution using the marginal distribution of the parent vertices, see Example 2.3.1

**Example 2.3.1 (Calculating marginal probability for $P(v_3s_1)$ given evidence to $v_2s_2$))**

| | | |
|---|---|---|
| $P(v_3s_1)$ | $=$ | $0.1 * 0.68 * 0$ |
| | $+$ | $0.2 * 0.68 * 1$ |
| | $+$ | $0.3 * 0.68 * 0$ |
| | $+$ | $0.6 * 0.32 * 0$ |
| | $+$ | $0.5 * 0.32 * 1$ |
| | $+$ | $0.4 * 0.32 * 0$ |
| $P(v_3s_1)$ | $=$ | $0.30$ |
| *or simply:* | | |
| $P(v_3s_1)$ | $=$ | $0.2 * 0.68$ |
| | $+$ | $0.5 * 0.32$ |
| $P(v_3s_1)$ | $=$ | $0.30$ |

In this manner, we need to evaluate all descendants as long as the parents' marginal probability is affected, and we have not given it evidence.

Updating the parent probability distribution is not as straightforward. Given that we know that $v_2s_2 = 1$, we need to evaluate the probability of each state of $v_1$. We use Bayes' rule for each of the parents' states, i.e. $P(v_1s_1|v_2s_2)$ and $P(v_1s_2|v_2s_2)$, see Example 2.3.2.

**Example 2.3.2 (Calculating probability of $P(v_1s_1|v_2s_2)$)**

| | | |
|---|---|---|
| $P(v_1s_1\|v_2s_2)$ | $=$ | $\frac{P(v_2s_2\|v_1s_1)P(v_1s_1)}{P(v_2s_2)}$ |
| $P(v_1s_1\|v_2s_2)$ | $=$ | $\frac{0.5*0.3}{0.5*0.3+0.1*0.7}$ |
| $P(v_1s_1\|v_2s_2)$ | $=$ | $0.68$ |

We propagate the evidence with Bayes' Rule to all ancestors and descendant that have no evidence. This cascading propagation of conditional probabilities can update the whole network when instantiating a single state. Updating the whole network is intractable for large networks with multiple arcs, which is a significant drawback to using BNs. It is therefore essential to limit this cascading propagation as much as possible.

#### 2.3.1.4 Causal networks

So far, we have only discussed the network in terms of vertices, arcs and probability distributions. If we want to use the network to reason about uncertainty, we can structure them to represent a causally connected network where the direction of the arcs indicates the direction of the causal relationship [Jen01]. In the initial network in Figure 2.3.2, we then say that $v_1$ causes $v_2$, and that $v_1$ and $v_2$ cause $v_3$.

For example, John needs to water his lawn with a sprinkler if it has not rained. John has a gardener that checks in the morning if he needs to turn on the sprinkler.

1. $v_1$ represents whether it rained
2. $v_2$ whether the sprinkler was on,
3. $(v_1, v_2)$ the causal effects of rain on the sprinkler's state.
4. $v_3$ represents whether the lawn is wet,
5. arcs $(v_1, v_3)$ and $(v_2, v_3)$ show the causal effect on the lawn.

**Definition 2.3.3 (D-separation [Jen01])**

> *Two distinct vertices $v_1$ and $v_2$ in a causal BN are D-Separated if for all paths between $v_1$ and $v_2$ there is an intermediate vertex $v_3$ (distinct from $v_1$ and $v_2$) such that either:*
>
> - *the connection is serial or divergent, and $v_3$ has evidence, or*
> - *the connection is convergent, and neither $v_3$ nor any of $v_3$'s descendants have evidence.*

When evidence cannot propagate between vertices, see Definition 2.3.3, it is called D-separation, and when it can propagate, it is called D-Connection.

Serial connection Vertices that are all consecutively connected along the same directed path are said to be serially connected, see Figure 2.3.4. We have removed the arc $(v_1, v_3)$, see Definition 2.3.6 for $v_3$'s values. $v_1$ and $v_3$ evidence affect the other two unless we instantiate $v_2$. In a serial connection, an instantiated vertex blocks propagation through that vertex, see Figure 2.3.4 Although $v_1$ has evidence, it does not affect $v_3$ because $v_3$ is only affected by the evidence in $v_2$. Additionally, $v_2$ has evidence and is also not affected.

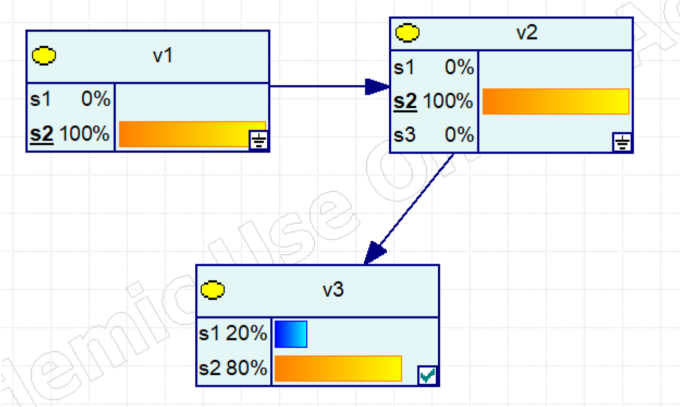
**Table 2.3.6** The conditional distribution defined for $v_3$ after the arc between $v_1$ and $v_3$ has been removed

| $v_2$ | $s_1$ | $s_2$ | $s_3$ |
|---|---|---|---|
| $s_1$ | 0.1 | 0.2 | 0.3 |
| $s_2$ | 0.9 | 0.8 | 0.7 |

For example, John walks out onto his lawn in the morning in his slippers. If it has rained $v_1$ during the night, the lawn will be wet $v_2$, and John will get his slippers wet $v_3$. Therefore if John's slippers become wet, he will reason that the lawn is wet and subsequently, given that the lawn is likely wet that it likely rained during the night. Conversely, if John knows that it rained during the night, he will expect the lawn to be wet and that if he walks on it in his slippers, then his slippers will become wet.

 33

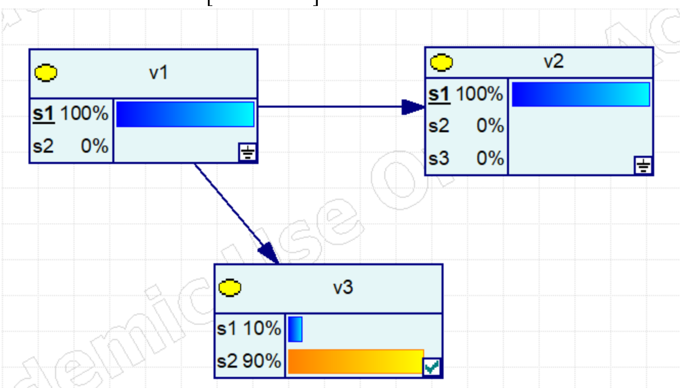
**Figure 2.3.4** A BN with an instantiated state, $v_2 s_2$, displayed in the GeNIe UI [DSL07a]




Conversely, if John knows that the lawn is wet, then information about whether it rained during the night will not affect his belief of whether his slippers get wet from walking on the lawn. Similarly, knowing that the lawn is wet, if his slippers become wet, it will not affect his belief of whether it rained during the night.

Divergent connection  A divergent connection is between sibling vertices, see Figure 2.3.5 where $v_2$ and $v_3$ are siblings, see Example 2.3.3. Evidence to one sibling affects the other sibling via their parent v1 unless the parent has evidence. Instantiating the parent in a diverging connection blocks inference between siblings. In Figure 2.3.5 both $v_1 s_1$ and $v_2 s_1$ have evidence; however, $v_3$ is only affected by $v_1 s_1$.

**Figure 2.3.5** A BN with an instantiated state, $v_2 s_2$, displayed in the GeNIe UI [DSL07a]




**Example 2.3.3 (The conditional distribution for $v_3$ after the arc between $v_2$ and $v_3$ has been removed)**

 34

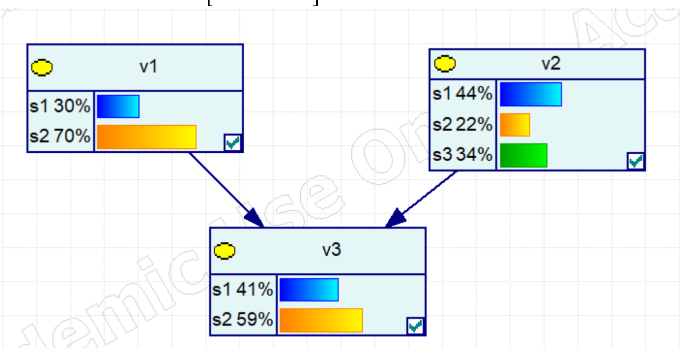
| $v_1$ | $s_1$ | $s_2$ |
|---|---|---|
| $s_1$ | 0.1 | 0.6 |
| $s_2$ | 0.9 | 0.4 |

For example, if John intends to mow the lawn $v_3$, then it matters to him whether it is wet $v_1$ since it is not good to mow the lawn if it is wet. If John walks out onto the lawn and his slippers get wet $v_2$, that will inform him that the lawn is wet, making it less likely that he will mow the lawn.

Conversely, if John already knows that the lawn is wet, then getting his slippers wet will no longer inform him of whether to mow the lawn or not. Similarly, if John already knows that the lawn is wet, then finding it not in good condition to be mowed will not inform him of whether his slippers get wet from walking on it.

Convergent connection A convergent connection is a connection between the parents of a child, see Figure 2.3.6. Evidence is not propagated through a convergent connection unless the child or its descendants have evidence.

**Figure 2.3.6** Convergent connection, displayed in the GeNIe UI [DSL07a]




If we give evidence to either parent, $v_1$ or $v_2$ , it will not affect the other parent, see Figure 2.3.7 evidence in $v_2 s_3$ affects $v_3$ but not $v_1$.

The marginal probability of $v_1 s_1$ remains the same because the conditional dependency is the same, see Table 2.3.7.

However, if we give evidence to a joint descendant, then the parents are affected by each other, see instantiated $v_3 s_1$ in Figure 2.3.8.

We now need the marginal probability of $v_2$ when evaluating $v_1$ because there is an altered conditional dependency between the two. Furthermore if $v_2$ has evidence, it will now affect the conditional probability of $v_1$ as seen in Figure 2.3.9, where $v_3 s_1$ and $v_2 s_3$ have evidence.

 35

**Figure 2.3.7** D-separation,
displayed in the GeNIe UI [DSL07a]

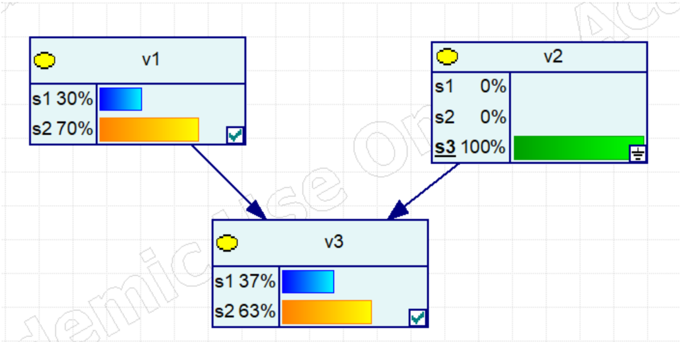



**Table 2.3.7** The marginal probability of $v_1 s_1$

| | | |
|---|---|---|
| $P(v_1 s_1)$ | $=$ | $P(v_1 s_1 \vert v_3 s_1) P(v_3 s_1)$ |
| | $+$ | $P(v_1 s_1 \vert v_3 s_2) P(v_3 s_2)$ |
| $P(v_1 s_1 \vert v_3 s_1)$ | $=$ | $\frac{0.1*0.3}{0.1*0.3+0.6*0.7}$ |
| $P(v_1 s_1 \vert v_3 s_1)$ | $=$ | $0.24$ |
| $P(v_1 s_1 \vert v_3 s_2)$ | $=$ | $\frac{0.9*0.3}{0.9*0.3+0.4*0.7}$ |
| $P(v_1 s_1 \vert v_3 s_2)$ | $=$ | $0.33$ |
| $P(v_1 s_1)$ | $=$ | $0.24 * 0.37$ |
| | $+$ | $0.33 * 0.63$ |
| $P(v_1 s_1)$ | $=$ | $0.3$ |

**Figure 2.3.8** D-separation where $v_3 s_1$ has been instantiated,
displayed in the GeNIe UI [DSL07a].

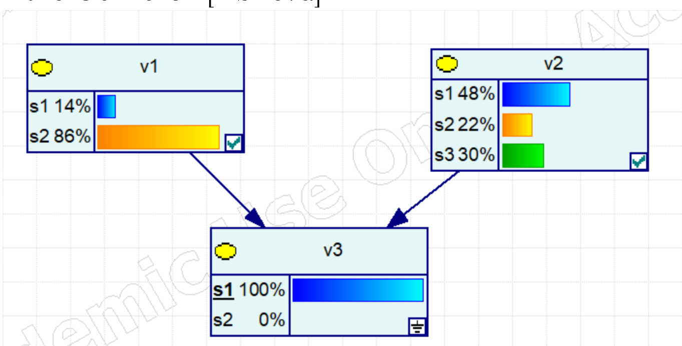
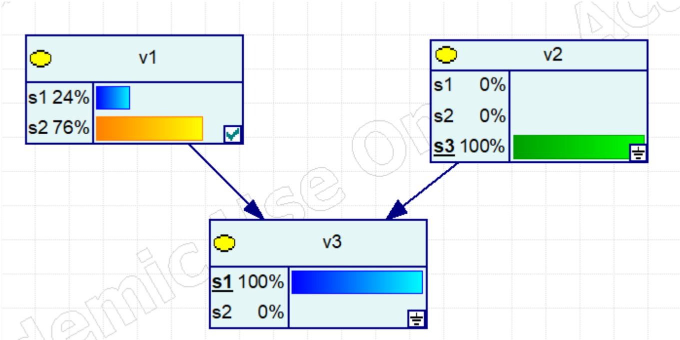



**Table 2.3.8** The conditional distribution for $v_3$, where $v_3 s_1$ and $v_2 s_3$ have been instantiated

| | | |
|---|---|---|
| $v_1$ | $s_1$ | $s_2$ |
| $v_2$ | $s_3$ | $s_3$ |
| $s_1$ | 0.3 | 0.4 |

**Figure 2.3.9** D-separation where $v_3s_1$ and $v_2s_3$ have been instantiated, displayed in the GeNIe UI [DSL07a]




In order to know $P(v_1s_1)$ we now need to take into account the changed conditional dependency, i.e. $v_1$ is now only dependant on the probability of $v_3s_1$ which is dependant on $v_2s_3$, see Example 2.3.4

**Example 2.3.4 (Calculating $P(v_1s_1|v_3s_1)$.)**

| | | |
|---|---|---|
| $P(v_1s_1)$ | $=$ | $P(v_1s_1\|v_3s_1)$ |
| $P(v_1s_1\|v_3s_1)$ | $=$ | $\frac{P(v_3s_1\|v_1s_1)P(v_1s_1)}{P(v_3s_1)}$ |
| $P(v_3s_1)$ | $=$ | $0.3*0.3+0.4*0.7$ |
| $P(v_3s_1)$ | $=$ | $0.37$ |
| $P(v_3s_1\|v_1s_1)$ | $=$ | $0.3$ |
| $P(v_1s_1)$ | $=$ | $0.3$ |
| $P(v_1s_1)$ | $=$ | $\frac{0.3*0.3}{0.37}$ |
| $P(v_1s_1)$ | $=$ | $0.24$ |

To understand this with the same example as before of John's wet lawn. The lawn could be wet ($v_3$) because it rained ($v_1$) or because the sprinkler went off ($v_2$). If John does not know whether the lawn is wet, then knowing whether it rained will not change his belief of whether the sprinkler went off, but it will affect his belief of whether the lawn is wet.

Conversely, if John knows that the lawn is wet, either directly or because his slippers got wet, then he will find it more likely that it either rained or that the sprinkler went off. If he then gets information that it did rain, then he may find it less likely that the sprinkler went off given that the lawn is wet.

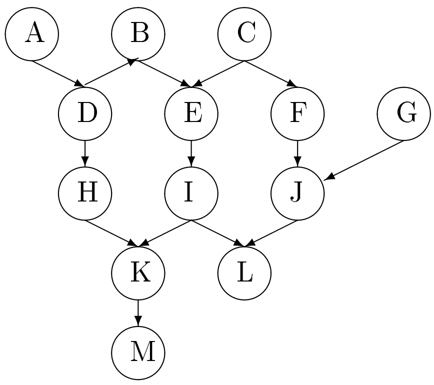
#### 2.3.1.5 Relevance Reasoning

D-separation is used to identify the part of the network to update after giving evidence to a vertex see Figure 2.3.10

**Example 2.3.5 (Relevance Reasoning)**

> *If we give evidence to A, it can travel to D, H, K, and M but not to I via K. It can also travel from D to B, E, I, and L, but not from L to J. From E it can travel to C, F, and J but it cannot travel to G.*
>
> *If we give evidence to B, then giving evidence to A will no longer affect B or any of the paths via B, it will only travel to D, H, K, and M.*
>
> *If we also give evidence to M, then it can travel on from K to I and from there to the rest of the network as before, except G, although it will neither affect B or M.*
>
> *Furthermore, if it is the case that we only need to know how a single or a few vertices, called target vertices, are affected by giving evidence to other vertices, that are distinct from the target vertices, then we can use D-separation to identify applicable paths. Using D-separation and target vertices in this manner is called Relevance Reasoning [LD97].*
>
> *For example, in the network in Figure 2.3.10, if we have instantiated B and M as before, but not subsequently evaluated the graph, and want to know what the value of H will be after instantiating A, then we need to update D, E, H, I and K.*

**Figure 2.3.10** A BN to demonstrate D-separation from [Jen01]




### 2.3.1.6 Object-Oriented Bayesian Networks (OOBNs)

BNs can quickly grow in size to become too large both in terms of space and computation. The conditional definition of each vertex grows exponentially with the number of its parents see Definition 2.3.4.

 38

**Definition 2.3.4 (The size of a vertex's conditional probability table (CPT))**

*Is the product of the number of its states N(s) and all parent states N(ps)*

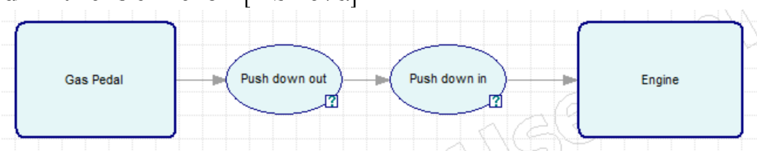
$$CPT(v_i) = N(v_i s) \prod_{p \in P} N(ps)$$

A vertex with too many parents can fill the computer's memory.

Moreover, the greater the coupling is between vertices the greater number of updates are required for any given evidence. We can use object-oriented architecture for BNs [KP97] by dividing the network into reusable objects. OOBNs are useful for large networks where we can reuse objects.

This process can be automated using D-separation; it can also follow logical reasoning. For example, see Figure 2.3.11, a car can be defined into well structured logical parts that can be divided further into sub-parts, a car engine has breaks, fuel consumption, steering, drive. The car body has doors, seats, and steering tools. For the most part, all of these are independent of each other except for a few well-defined functions. Stepping on the gas pedal increases the fuel intake but has no effects on the breaks or the steering. We only need to update the fuel intake when stepping on the gas. Subsequently, increased fuel will speed up the engine, which in turn will cause other parts to be affected. We model each part into object BNs, called submodels, with in- and out-vertices. In our example, the gas pedal increases the fuel percentage of the out-vertex, and the engine's in-vertex feeds the effects from the pedal to the engine.

**Figure 2.3.11** An example of a pedal and engine using OOBN, displayed in the GeNIe UI [DSL07a].




The two submodels in Figure 2.3.11 are connected, but that is not necessary, they can be independent BNs and only loaded and updated on demand.

#### 2.3.1.7 Decision Networks

We use BNs as decision networks and extend them with utility and decision vertices. Utility vertices:

- Define the utility of potential outcomes.
- Utility $U = u_1, ..., u_N$ where N is the total number of utility vertices.
- Are blue diamonds in the GeNIe UI [DSL07a].
- Can have decision and chance vertices as parents.
- Cannot be parent vertices.

A decision vertex:

- Can have a list of actions, decisions or goals for evaluation.
- We call the states, of the decision vertex, strategies $= (\sigma_1, ..., \sigma_I)$ where I is the number of strategies in a decision vertex.
- Are green rectangles in the GeNIe UI [DSL07a].
- Can be a parent of both utility and chance vertices.
- Cannot have parent vertices.

**Definition 2.3.5 (Expected utility [Jen01])**

*EU $EU(\sigma_i)$ for $i \in I$ and for each utility $u_n \in N$ of a chance vertex $v \in V$ and its conditional probability given the evidence,*

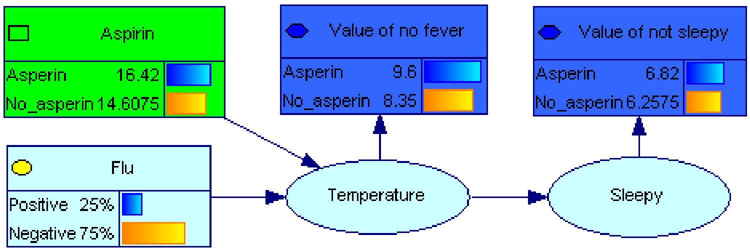
$$EU(\sigma_i) = \sum_{n=1}^{N} \sum_{v \in V} u_n(v) P(v|\sigma_i, e)$$

A combination of decision vertex and utility vertices can calculate the expected utility (EU), see Definition 2.3.5 for how to calculate EU:

- If a decision vertex is a parent of a chance vertex, then the chance vertex defines the strategies' probability.
- If a decision vertex is a parent of a utility then the utility defines the cost of a given strategy $\sigma$.
- A utility defines the cost of parent strategies.
- If a utility vertex has chance vertices as parents, then the utility defines a value for each state of the chance vertex.
- A utility calculates the EU, given the strategies and the probability of those strategies being true.

The following example shows how to use decision BNs. We assume that in Iceland, the general population has the flu about one quarter of the year. The flu increases the probability of fever, and having a fever increases the probability of being sleepy. Moreover taking Aspirin is very likely to reduce the fever to average temperature, 37°C. The Icelandic government has taken the initiative to distribute free Aspirin to every Icelandic home to increase the small population's productivity and that taking Aspirin has no side effects.

**Figure 2.3.12** The Flu example of a Decision Network, displayed in the GeNIe UI [DSL07a]




In Figure 2.3.12 is the Bayesian net showing the causal connections from flu to fever to sleepy,

The Aspirin decision vertex has two strategies:

- $\{Aspirin = to\ take\ Aspirin\}$
- $\{No\_Aspirin = not\ to\ take\ Aspirin\}$

The flu chance vertex has two states: $\{Positive\}$ with a probability of 0.25 and $\{Negative\}$ with a probability of 0.75.

**Table 2.3.9** The Temperature chance vertex for the Flu example

| Aspirin | Aspirin | | No Aspirin | |
|---|---|---|---|---|
| Flu | Positive | Negative | Positive | Negative |
| Normal | 0.9 | 0.98 | 0.4 | 0.98 |
| Elevated | 0.1 | 0.02 | 0.6 | 0.02 |

See Table 2.3.9 for the temperature vertex. The *Temperature* chance vertex has two states:

- $\{Normal,\ the\ temperature = 37°C\}$,
- $\{Elevated,\ the\ temperature\ > 37°C\}$

**Table 2.3.10** The Sleepy chance vertex for the Flu example

| Temperature | Normal | Elevated |
|---|---|---|
| Yes | 0.3 | 0.75 |
| No | 0.7 | 0.25 |

See Table 2.3.10 for the sleepy chance vertex. In Figure 2.3.12, we can see from the two utility vertices that the value of taking Aspirin to reduce fever is 9.6, and the value of taking Aspirin to reduce sleepiness is 6.82.

**Table 2.3.11** Utility vertices for the Flu example

Value of no fever

| Temperature | Elevated | Normal |
|---|---|---|
| value | 0 | 10 |

Value of not sleepy

| Sleepy | yes | no |
|---|---|---|
| value | 0 | 10 |

See Table 2.3.11 for both utility values.Therefore, the strategy to take Aspirin carries a gain of $9.6 + 6.82 = 16.42$, and to not take Aspirin carries the gain of $8.35 + 6.2575 = 14.6075$.

In Figure 2.3.13, the Flu vertex has the $Flu, Positive$ instantiated. In Figure 2.3.14 the Flu vertex has the value of the $Flu, Negative$ instantiated.

**Figure 2.3.13** In the Flu example $Flu, Positive$ is instantiated, displayed in the GeNIe UI [DSL07a]

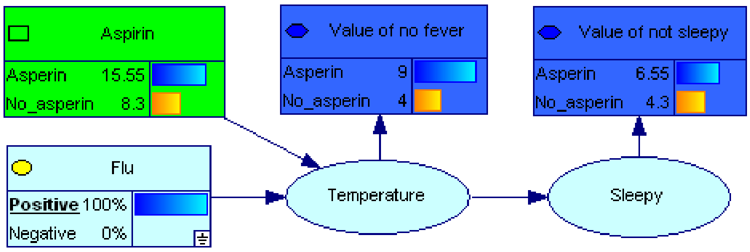
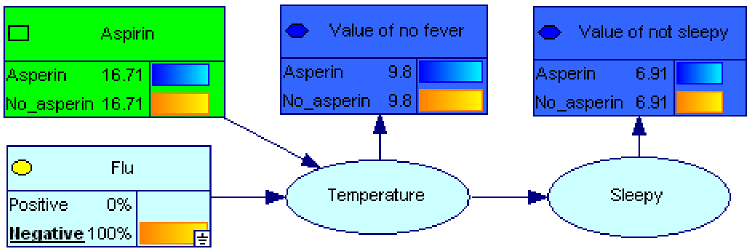



**Figure 2.3.14** In the Flu example $Flu, Negative$ is instantiated, displayed in the GeNIe UI [DSL07a]




It is useful to instantiate vertices to understand the values that would cause such extreme and how it would affect other vertices and evaluate the strategies. In Figure 2.3.13, the assumption is that the individual has flu, and it is then clearly evident that there is a higher gain in taking Aspirin since it yields a higher value, 15.55 to 8.3. In Figure 2.3.14, the assumption is that the individual does not have any flu, and then there is no higher gain in taking Aspirin than not taking Aspirin. Therefore, in this example, the expected utility indicates a gain of taking Aspirin than not taking it.

 42

### 2.3.2 Game Theory

This Section describes the essentials of GT that are necessary to understand the problem domains tackled by this thesis and the solution proposed. GT is a mathematical definition of rationality.

#### 2.3.2.1 Terminology

A game in GT:

- Has a set of two or more players, $(p_i, ..., p_{\mathcal{I}}$ where $\mathcal{I}$ is the number of players.
- A set of strategies $s_k, ..., s_{\mathcal{K}}$ where $\mathcal{K}$ is the number of strategies per player.
- A set of strategies for a player is called the player's strategy profile.
- A strategy profile $\sigma_j, ..., \sigma_{\mathcal{J}}$ where $\mathcal{J}$ is the number of strategy profiles.
- A distinct strategy played with the probability of 1 is called a pure strategy.
- A mixed strategy is a strategy of two or more distinct strategies that are all played depending on a probability distribution. For example $(s_1 = 0.5, s_2 = 0.5)$ means that $s_1$ is played half the time and $s_2$ the other half.
- A $\sigma_j$ can contain both pure and mixed strategies.
- Each player has a utility function $U$ for each of the strategy profiles that he can play.

We introduce how a game in GT works in Aumann's example [FT91], see Example 2.3.6.

**Example 2.3.6 (Aumann's example [FT91])**

*Player 1 and 2 want to find an optimal strategy in a game where they both have equal expected utility.*

| | | $p_2$ | |
|---|---|---|---|
| | | $s_1$ | $s_2$ |
| $p_1$ | $s_1$ | *5,1* | *0,0* |
| | $s_2$ | *4,4* | *1,5* |

In Example 2.3.6 there are two players $(p_1, p_2)$ and two strategy profiles each with two strategies: $p_1, S = (s_1, s_2)$ and $p_2, S = (s_1, s_2)$.

If $p_1$ plays $s_1$ and $p_2$ plays $s_1$ then $p_1$'s $U$ is 5, denoted $U(p_1(s_1, s_1)) = 5$, and $p_2$'s $U$ of 1, denoted $U(p_2(s_1, s_1)) = 1$. We calculate the EU for a strategy profile $\sigma_i$ by aggregating the utility of each strategy weighted by the probability distribution of the players and opponents $\sigma_i$, see Definition 2.3.6 and see Example 2.3.7.

**Definition 2.3.6 (EU for a player's strategy profile [FT91])**

*$\sigma_j, ..., \sigma_{\mathcal{J}}$ are all the strategy profiles.*

$$EU(\sigma_i) = \sum_{s \in S} \left( \prod_{j=1}^{J} \sigma_j(s) \right) U_i(s) \tag{2.1}$$

For example, in a mixed strategy, the *EU* is dependent on the probability distribution of $S$ for both player and opponent. In Example 2.3.6, if the players play mixed strategies with each strategy with 0.5 probability, $P(s_i) = 0.5$, the *EU* is 2.5, see Example 2.3.7.

**Example 2.3.7 (Calculating EU for the game in Example 2.3.6)**

*EU for $p_1$*

| | | | | | | |
|---|---|---|---|---|---|---|
| $EU(p_1 s_1)$ | = | $0.5 * 0.5 * p_2 s_1$ | + | $0.5 * 0.5 * p_2 s_2$ | | |
| $EU(p_1 s_1)$ | = | $0.5 * 0.5 * 5$ | + | $0.5 * 0.5 * 0$ | = | 1.25 |
| $EU(p_1 s_2)$ | = | $0.5 * 0.5 * p_2 s_1$ | + | $0.5 * 0.5 * p_2 s_2$ | | |
| $EU(p_1 s_2)$ | = | $0.5 * 0.5 * 4$ | + | $0.5 * 0.5 * 1$ | = | 1.25 |
| $EU(p_1 S)$ | = | *1.25* | + | *1.25* | | |
| $EU(p_1 S)$ | = | *2.5* | | | | |

*EU for $p_2$*

| | | | | | | |
|---|---|---|---|---|---|---|
| $EU(p_2 s_1)$ | = | $0.5 * 0.5 * p_1 s_1$ | + | $0.5 * 0.5 * p_1 s_2$ | | |
| $EU(p_2 s_1)$ | = | $0.5 * 0.5 * 1$ | + | $0.5 * 0.5 * 4$ | = | 1.25 |
| $EU(p_2 s_2)$ | = | $0.5 * 0.5 * p_1 s_1$ | + | $0.5 * 0.5 * p_1 s_2$ | | |
| $EU(p_2 s_2)$ | = | $0.5 * 0.5 * 0$ | + | $0.5 * 0.5 * 5$ | = | 1.25 |
| $EU(p_2 S)$ | = | *1.25* | + | *1.25* | | |
| $EU(p_2 S)$ | = | *2.5* | | | | |

Each game has an equilibrium, see Definition 2.3.7.

**Definition 2.3.7 (Equilibrium [FT91])**

*Equilibrium is the strategy that has equal or higher EU to any other EU in the game for all players.*

We find equilibrium by evaluating the strategies and their probability distribution.

An *opponent* in GT is:

- A common denominator for other players.
- Does not indicate competition or a competitor.
- Is any other player that is not the player.
- Can be cooperative or non-cooperative.
- Is denoted with $-i \in \mathcal{I}$ where $-i \neq i$.

**Definition 2.3.8 (Rationality [FT91])**

> *Rationality is choosing the strategy that will give maximum EU no matter what the opponents play, given current information.*

Every player that plays rationally, see Definition 2.3.8, always chooses the equilibrium strategy. If the player does not act rationally, then they are considered to be making a mistake. For example, if they push a wrong button because their hands are trembling.

There are various levels of information that a player can have in-game:

- A *completely informed* player is someone who knows all the possible utility functions, and strategy profiles and everyone knows that he is completely informed and he knows that everyone knows that he is completely informed, and so on.
- A game is a *completely informed game* if all the players are completely informed.
- An *incompletely informed* player lacks information about their opponents' utility functions and strategy profiles.
- An *imperfectly informed* player is an incompletely informed player that can, by calculation, estimate the strategy profiles and EUs of opponents.
- To reason under uncertainty is when a player estimates the strategy profiles and EUs of opponents.

A *non-cooperative* game is when the players have no prior binding agreements. In most realistic scenarios, the players do not have prior signed contracts with their opponents.

#### 2.3.2.2 Nash equilibrium

A Nash equilibrium, see Definition 2.3.9 is a profile of strategies such that each player's strategy is an optimal response to each of the other players strategies, i.e. there is no incentive to deviate.

**Definition 2.3.9 (Nash Equilibrium [FT91])**

> *A mixed-strategy profile $\sigma^*$ is a Nash equilibrium if for all players $i$, and for all strategies $s_i \in S_i$,*
>
> $$EU(\sigma_i^*, \sigma_{-i}^*) \geq EU(s_i, \sigma_{-i}^*) \tag{2.2}$$

Nash equilibrium holds for the *extended* game, i.e. when playing repeatedly against the same set of players, as proven by J. Nash [Nas51]. There are three Nash equilibriums in Example 2.3.6. Two pure strategies $(p_1s_1, p_2s_1)$ and $(p_1s_2, p_2s_2)$ and one mixed strategy $((0.5{*}p_1s_1, 0.5{*}p_1s_2), (0.5{*}p_2s_1, 0.5{*}p_2s_2))$. In Example 2.3.8 we find the first equilibrium.

**Example 2.3.8 (Finding Nash Equilibrium)**

> *When finding Nash Equilibrium we can try every strategy until we find the set with maximum value. In this example we will find the pure strategy equilibrium (U,L).*
>
> 1. *$p_1$ plays $s_1$ and can see that $p_2$'s optimal response is $s_1$ since $U(p_2s_1) > U(p_2s_2)$, $1 > 0$.*
> 2. *$p_1$'s $U(p_1s_1,p_2s_1) = 5$ and is $p_1$'s current max.*
> 3. *$p_1$ plays $s_2$ and can see that $p_2$'s optimal response is $s_2$ since $U(p_2s_2) > U(p_2s_1)$, $5 > 4$.*
> 4. *$p_1$'s $U(p_1s_2,p_2s_2) = 1$ and $1 < 5$ so $p_1$'s optimal strategy and Nash equilibrium is $s_1$.*

#### 2.3.2.3 Bayesian equilibrium

Harsanyi proposed Bayesian equilibrium to calculate Nash equilibrium for incomplete information games, where players lack complete information about the game world [Har95], [FT91].

In J. C. Harsanyi's words:

> In our own view it has been a major analytical deficiency of existing GT that it has been almost completely restricted to

> complete information games, in spite of the fact that in many real-life economic, political, military, and other social situations the participants often lack full information about some important aspects of the "game" they are playing [Har67] p:163:.

An incomplete information game is where some players lack full information about the game. We assume that each player knows their utility function and strategy profiles.

For example, two warring factions; that know their state but not the complete state of their opponent. They will have some intelligence of their opponents' states and can reason about them.

We can assume that there are finitely many states for the opponent to be relevant to the player. For example, the opponent has a secret weapon, depleted their food or resources, or has powerful tanks.

The players do not know their opponents' current utility function but know all utility functions that the opponents could be using, which is exogenous data because it is:

- Not derived from the opponents' status.
- Is a factor in the equilibrium-calculations. For example, strategic places, pay off functions, and physics.
- Is considered common knowledge among the players.
- Stands outside the influence of equilibrium calculations.
- Is constant throughout the game.

**Player-centred Bayesian equilibrium** In a player-centred Bayesian equilibrium the player reasons about what the opponent's utilities functions are and what the opponent could do in response to their actions, which leads to an infinite recursion as follows:

- A player's first order expectations $P_i^1$ is a subjective probability distribution, $P_i^1(U_{-i})$, over all possible opponent utilities, $U_{-i}$.
- A player's second-order expectations is a subjective $P_i^2(P_{-i}^1)$.
- A player's $k$th-order expectations is a subjective $P_i^k(P_{-i}^{k-1})$

Therefore, the player-centred game contains infinite recursion that requires a fixed point configuration to calculate equilibria [Har67].

**Type-centred Bayesian equilibrium** An alternative to player-centred Bayesian equilibrium is an equivalent type-centred Bayesian equilibrium, that does not lead to infinite recursion, and is structured as follows:

- A player has a finite set of possible types $\theta_i \in \Theta$ that represent every possible makeup of the player.
- $\theta_{-i} = (\theta_1, ..., \theta_{i-1}, \theta_{i+1}, ..., \theta_I)$. is the opponents type.
- A $\theta_i$ can represent the players character, beliefs of other player's strategy profiles, and utility functions.
- A $\theta_i$ is any prior information that is not common knowledge and that may affect the $U_{-i}$, their decision making and their belief about $\theta_{-i}$.
- Some players may be better informed than others about their $\theta_{-i}$.
- The exogenous data of the game is all $\theta \in \Theta$.

Rather than reason about all possible utility functions and all possible opponents' possible strategies, we can use the types to find a Bayesian equilibrium [Har67, Har68a, Har68b]. If the player does not know the $\theta_{-i}$, it can find the probability distribution of the $\theta_{-i} \in \Theta$.

- A $\theta_i \in \Theta$ has an objective prior probability distribution $P(\theta_i)$.
- $P(\theta_{-i}|\theta_i)$ is the conditional probability of the $\theta_{-i}$ given the $\theta_i$.
- A player can use Bayes' rule in ex-ante calculations to evaluate the $P(\theta_{-i}|\theta_i) \in \Theta$, hence the name Bayesian equilibrium.

When a player knows the probability distribution over all $\theta_{-i} \in \Theta$, they can use this to evaluate the EU of the opponent given $\theta_i$.

- $\sigma_i(\theta_i)$ is the player's strategy profile which is dependant on its $\theta_i$, see Definition 2.3.10.
- Given the $\sigma_i(\theta_i)$, player can know the set of opponent's strategies $s_{-i}\sigma_{-i}$.
- A player can calculate the EU$(\sigma_{-i}(\theta_{-i}))$ $P(\theta_{-i}|\theta_i)$ if the player knows $s_{-i}\sigma_{-i}$.

### Definition 2.3.10 (Type-centric strategy profile [FT91])

*Given a strategy profile $s(\cdot)$, and an $s'_i(\cdot) \in S_i^{\Theta_i}$, let $(s'_i(\cdot), s_{-i}(\cdot))$ denote the profile where player i plays $s'_i(\cdot)$ and the other players, his opponents, play $s(\cdot)$, we let*

$$(s'_i(\theta_i), s_{-i}(\theta_{-i})) = (s_1(\theta_1), ..., s_{i-1}(\theta_{i-1}), s'_i(\theta_i), s_{i+1}(\theta_{i+1}), ..., s_i(\theta_i))$$

We can find Bayesian Equilibrium based on ex ante calculation of $P(\theta_{-i}|\theta_i)$ and exogeneous knowledge of the $s_{-i}\sigma_{-i}\theta{-i}$, see Definition 2.3.11

**Definition 2.3.11 (Bayesian Equilibrium using player types [FT91])**

*We find an equilibrium by maximizing the sum of player i's $P(\theta_{-i}|\theta_i)EU(s_i')$ for each of its strategies $s_i' \in S_i$.*

$$s_i(\theta_i) \in \arg\max_{s_i' \in S_i} \sum_{\theta_{-i}} P(\theta_{-i}|\theta_i) EU(s_i', s_{-i}(\theta_{-i}), (\theta_i, \theta_{-i})) \quad (2.3)$$

#### 2.3.2.4 Multi-Agent Influence Diagram

Koller and Milch [KM03] propose a MAID to reason under uncertainty. A MAID is an extended Bayesian net with:

- A set of players $p_i \in \mathcal{I}$, where $\mathcal{I}$ is the number of players.
- A set of utility vertices, $\mathcal{U} = \{U_1, ..., U_m\}$.
- A decision vertex $\mathcal{D}$ with the set of player strategies $s_i \in \mathcal{S}$ .
- A set player's strategy profiles $\sigma_i \in \sigma^*$.
- An opponent's strategy profile $\sigma_{-i}$.
- A subset of $s_i$ is $\varepsilon \subset \mathcal{S}$.
- A strategy profile over a $\varepsilon \subset \mathcal{S}$ is
- $\sigma_{-\varepsilon}$ is a $\sigma$ that does not include any $s \in \varepsilon$.

A MAID is a causally structured BN with $\sigma_i$ that is given evidence from a $\sigma_{-i}$ in order to calculate the EU($\sigma_i$), this is called strategic relevance.

We can calculate the expected utility (EU) as defined for a BN, see Definition 2.3.5, adjusted for a $\sigma_i$, see Definition 2.3.12.

**Definition 2.3.12 (Expected Utility for $\sigma_i$ in a MAID [KM03])**

*If $\mathcal{M}$ is a MAID and $\sigma$ is a strategy profile for $\mathcal{M}$, then the joint distribution over $\mathcal{V}$ defined by the Bayesian net is $P_{\mathcal{M}[\sigma]}$, denoting the joint distribution for $\mathcal{M}$ induced by $\sigma$. Suppose that $\mathcal{U} = \{U_1, ..., U_m\}$.*

$$EU_a(\sigma) = \sum_{U \in \mathcal{U}_a} \sum_{u \in dom(U)} P_{\mathcal{M}[\sigma]}(U = u) \cdot u \tag{2.4}$$

We can now calculate EU($\sigma_i \in \sigma^*$) and find the set of maximum EU($\sigma_i$), equivalent to finding Nash Equilibrium, see 2.3.13

**Definition 2.3.13 (MAIDs equilibrium [KM03])**

*We say that $\sigma_{-\varepsilon}$ is optimal if for $\sigma^*_\varepsilon$ it is equal or greater in MAID $\mathcal{M}[\sigma_{-\varepsilon}]$, than for any other $\sigma_{-\varepsilon}$:*

$$EU_a(\sigma_{-\varepsilon}, \sigma^*_\varepsilon) \geq EU_a(\sigma_{-\varepsilon}, \sigma^{'}_\varepsilon) \tag{2.5}$$

The MAIDs equilibrium is equivalent to finding Nash Equilibrium in an imperfectly informed game, see Definition 2.3.9 and below for ease.

$$EU(\sigma_i^*, \sigma_{-i}^*) \geq EU(s_i, \sigma_{-i}^*) \tag{2.6}$$

Furthermore, the MAIDs equilibrium uses BNs to create MAID $\mathcal{M}[\sigma_{-\varepsilon}]$ for a $\sigma_i$ equivalent to a type-centric Bayesian equilibrium, see Section 2.3.2.3 and in Definition 2.3.11.

The MAIDs are therefore using the same approach as with Bayesian type-centric equilibrium to find a Nash equilibrium for players reasoning under uncertainty using causally connected BNs.

### 2.3.3 Summary

We introduced the computational techniques that we use for DED

1. Bayesian Network (BN)'s:
   (a) Causal network structure BNs which we use for coherent agent reasoning
   (b) OOBN architecture, which we use to manage DED agents' knowledge bases
   (c) D-separation, which we use to optimise computation for the agents to respond within one second.
2. Game Theory (GT):
   (a) To have a mathematical definition of rationality.
   (b) Introduce Bayesian Equilibrium for rationality under uncertainty.
   (c) MAIDs to combine causally structured OOBN with GT

The MAIDs are at the core of our contribution to more believable characters in terms of fluid, coherent and rational responses.

## 2.4 Former work

We build the DED on two former work, the Dynamic Plot Generating Engine (DPGE), see Section 2.4.1 which dynamically creates novel mystery plots from static input, and the RDE, see Section 2.4.2, which uses the MAIDs to create rational agents.

### 2.4.1 Dynamic Plot Generating Engine

The Dynamic Plot Generating Engine is F. Arinbjarnar's BSc final project, in 2004, at Reykjavik University (RU), Iceland.

It was a solo research project, in part guided by a visiting professor Dr Finn Verner Jensen from Aalborg University, Denmark. Dr Jensen had worked on a similar project, *Non-Linear Interactive Storytelling Using Object-Oriented BNs (Nolist)* [BJJ+04].

The DPGE uses Propp's Morphology, previously introduced in Section 2.2.1.3, and causally structured BNs, previously introduced in Section 2.3.1, to generate a nearly endless supply of plots to use in mystery based computer games.

#### 2.4.1.1 The plot

A story begins at some point in time, which is understood to have a prior narrative, which is frequently implicit and left to the reader's imagination. The mind instantly creates a backstory for all of what it senses; even the most uncomplicated narrative will have a former world state. The mind will draw a picture of a plausible scenario and a narrative.

The implicit necessity of previous events makes us see a narrative where there is no action; even in a static picture there:

> " *...is a shadowy sense of time preceding it, and specifically of narrative time - that is, time comprised of necessary events that lead up to and accounts for, what we see .* "
>
> – [Abb02, p. 6]

In murder mysteries, this previous state, the past, is the sequence of events that lead up to and cause the murder, i.e. the murder plot. Conversely, the mystery proper is the narrative of the sleuth's murder investigation. There are, therefore, two distinct plots in a Murder Mystery [Tod77]:

1. The murder plot that has occurred before the mystery proper starts.
2. Present, the narrative of the sleuth uncovering the murder plot.

The DPGE generates the first plot, the murder plot, at the mystery's start to provide auto-generated mystery plots for murder mystery games.

Conversely, in Nolist [BJJ+04], the engine creates the murder plot during the player's investigation. Nolist will create the murder plot depending on how the player plays the game, such that, if the player finds a body and a gun nearby, the probability that the murder victim was shot with a gun increases if the player finds a gun nearby. Similarly, the probability knife stabbing increases if the player finds a knife before finding the gun.

Dr Jensen was aware of the two plots, and we discussed why he chose this approach [Jen04]. He feared that smart players would find it too easy to solve the mystery plots, quickly identify the murderer and subsequently miss out on the intended drama. Conversely, that the difficulty of solving the mystery would confound other, not as sharp, players. In both cases, this could lead to boredom, and players would miss out on the drama. For this reason, the engine would adjust the murder plot to ensure that the player needed to explore the world and interact with the characters to a certain degree to solve the mystery. Similar to a teacher giving more capable students more challenging work.

It is imperative to address the challenges of combining game mechanics and the narrative; however, there are alternative methods to control the difficulty levels of games; additionally, it is difficult to conjecture how much this particular element will affect the drama's quality. It is essential to consider the player's agency and how much control we give to the player. It is, arguably, a design decision whether to block smart players from finishing quickly.

#### 2.4.1.2 The structure

The DPGE combines BNs and Propp's formalism, introduced in Section 2.2.1.3, to generate the plots. Propp defined four rules; the DPGE generates plots that satisfy them as follows:

1. "Functions of characters serve as stable, constant elements in a tale, independent of how and by whom they are fulfilled. They constitute the fundamental components of a tale"[Pro68].
   The mystery morphology has a finite set of character functions for motive, means, opportunity and murder; for the three character types: victim, murderer and suspect.
2. "The number of functions known to a fairy tale is limited"[Pro68].
   The mystery morphology has a finite set of 4 functions listed in #1.
3. "The sequence of functions is always identical"[Pro68].
   We create the mystery in a causally structured BN; the causal structure implies a consistent sequence.
4. "All fairy tales are of one type in regard to their structure"[Pro68].
   We generate all mysteries with the same algorithm, and they are all of the same structure.

As an example of how the BN is defined here are some of the motive functions:

1. **Swindle.** Either the murderer is swindling the victim, and the victim threatens to tell, or the victim was swindling the murderer, and the murderer wanted revenge. Either way, the swindler should be rich.
2. **Blackmail.** Either the murderer was blackmailing the victim and feared that the victim would reveal it or the victim was blackmailing the murderer. In either case, it is necessary that the blackmailed character has some dark secret to blackmail.
3. **Wedlock.** In Agatha Christie's time, it could sometimes be difficult or bad publicity to get a divorce and found it justifiable to murder their partner instead. Wedlock can apply today if people do not wish to go through the divorce process and lose half their belongings and possibly custody of their children. The constraint in the net for wedlock is that the murderer is the victim's partner.
4. **Inheritance.** is self-explanatory; the victim must be rich, and the murderer must be their heir.
5. **Adultery.** The victim was having an affair with the murderer's spouse, and the murderer kills because of pure jealousy.
6. **Revenge.** The murderer believes that they have cause for revenge. The victim must have harmed the murderer in some sense in the past.
7. **Debt.** The murderer owed the victim lots of money and could not pay, i.e. the murderer is poor.

#### 2.4.1.3 Dynamically generated plots

The DPGE generates the plots *dynamically* using an abstract representation of the game world in the form of a causally structured BN that is read in at the start and creates a novel plot by randomly instantiating the BN's chance vertices. For this reason, it is useful for generating a constant stream of novel plots for games.

Step 1, Reading data, the engine reads in a text file that details all possible names of persons used by the engine. Next, the engine then reads a text file that details all variable types in the net and their connection to other variables.

Step 2, Drawing the net, each variable in the BN is programmatically added; using the C++ library Smile. [DSL07b].

**Figure 2.4.1** Instantiating a murderer in the BN, displayed in the GeNIe UI [DSL07a]

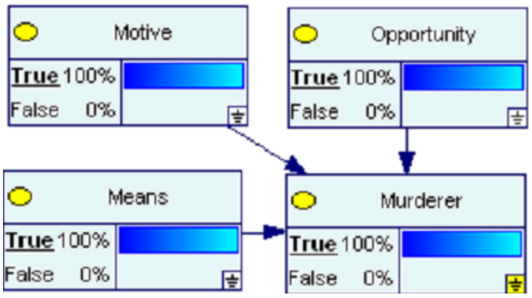
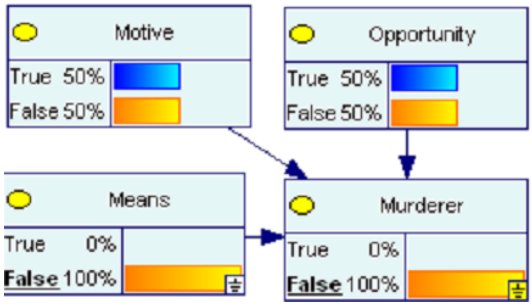



**Figure 2.4.2** Instantiating a suspect, displayed in the GeNIe UI [DSL07a]




Step 3, Instantiating specific variables, the next step is to decide on a murderer by randomly choosing one character. A murderer needs to have a motive, means and opportunity so these variables, in the murderer subnet, need to be instantiated to true, see Figure 2.4.1.

Similarly, to eliminate a suspect as a possible murderer, the suspect can not have all three: motive, means and opportunity. We, therefore, instantiate one of them, randomly, to false for each of the suspects. For example, see Figure 2.4.2, the variable means has bean instantiated to false.

Step 4, Instantiating the plot, finally, we instantiate all remaining variables by generating a random number between 0 and 1 and select the state that corresponds to its value.

This dynamic process creates novel plots regardless of whether the input is static. For example;

1. **Static input**, when a player starts a new game, a new plot is generated for the player using the DPGE. This new plot will be significantly different from previous plots.
2. **Semi-static input**, at the start of the game, the player can choose some of the game world's states. For example, their background and role. These user choices instantiate corresponding nodes in the BN to reflect the bespoke requests of the player.

55

3. **Dynamic input**, the DPGE can further be used after the player has solved one mystery to create new mysteries from the updated state of the BN; as a result of the player's gameplay.

#### 2.4.1.4 Tests

The engine is easily connected to other applications and tested in diverse setups and requirements.

The first test of the engine's applicability was to create a simple mystery game that used plots generated by DPGE.

The mystery game provided the player with predefined logical conclusions based on the plot and showed that each plot was consistent and that the murderer and the murder weapon was deducible. The game demonstrated that each new plot generated was distinct and complete.

We connected the engine to the Hexia dialogue control system [13]. Hexia provided means to interact with the engine through MSN Messenger. This setup provided an interactive mystery where the player could talk with an NPC named Theresa and ask her to perform actions on the scene such as "open drawer", "read letter" and "look at desk".At the beginning of each game, Theresa greeted the player and described the scene. The player then needed to tell Theresa what to do next, and she then told the player the results of her actions.

For example, the player could have asked Theresa to look at a desk, and Theresa then informs the player that there is a drawer in the desk. The player can then ask Theresa to open the drawer. The player does not need to indicate which drawer as the Hexia dialogue control system is contextual; it can preserve context in a dialogue. Theresa will then attempt to open the drawer, and if she is successful then she will describe the content of the drawer; otherwise, she will describe the problem, e.g. "the drawer is locked!" At any given time the player could frame a suspect and get informed by the engine whether she was correct or not. The player can only frame suspects that she has become aware of in the game.

We demonstrated this implementation of the engine at the Reykjavik Artificial Intelligence festival (2006). Those who tried it were successful in playing the game and deducing who the murderer was after a few attempts. It should be stated that no formal tests were done that would prove this. We did no formal tests to prove this.

Authoring the dialogue for NPC Theresa took less than three 40 hour work weeks.

### 2.4.2 Rational Dialogue Engine

The RDE creates and runs a set of actors that are rational in Game-Theoretic terms, i.e. choose their optimal strategy, i.e. the strategy that has the highest value as explained in Section 2.3.2, that can engage in a dialogue with each other and the player, and. The RDEwas developed as part of F. Arinbjarnar's MSc thesis [Ari07a] and published in *AAAI Fall Symposium On Intelligent Narrative Technologies* [Ari07b].

In RDE, a rational actor plays a character that interacts with other characters in a computer game played by human players or other actors. We define the actor's rationality as the choice of actions which best satisfy the character's objectives and desires. We base the character's objectives on what would typically motivate a person [HHL$^+$92], i.e. believable goals. For example, in a murder investigation, the detective has the objective to find the murderer, while other characters have the objective not to reveal a motive, means or anything else that could make them a likely murderer.

**Rational actors** We structure the rational actor in RDEwith three distinct parts:

1. A past life for the character to build a knowledge base that contains information of their present state and their knowledge of other characters in the game. Essentially, the DPGE introduced in Section 2.4.1 creates a plot by instantiating a causally structured BN, see Section 2.3.1 for an introduction to BNs.
2. Each actor has a knowledge base structure by their past life, i.e. the plot generated by the DPGE. Therefore, their knowledge base is a causally structured BNs that contains the character's knowledge in the form of instantiated chance vertices in their BNs. This knowledge base is updated as the character gains new knowledge.
3. A reasoning algorithm that can find a game-theoretic equilibrium as explained in Section 2.3.2, and uses the BNs from their knowledge base to reason about the current state of the game world.

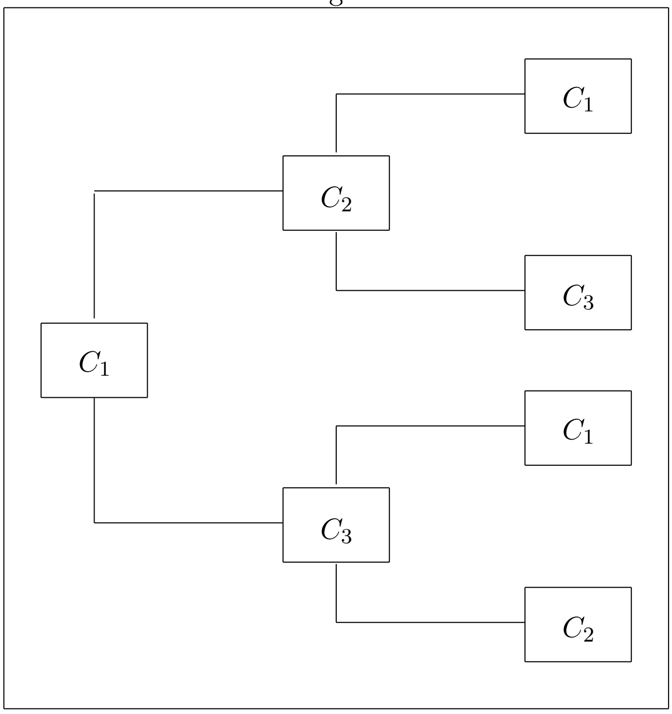
#### 2.4.2.1 Character's knowledge base

The actor has a set of knowledge subnets where it maps everything its character knows. When the actor is reasoning about how to respond in a dialogue, it requires a mental image of its opponent's knowledge with a game-theoretic game.

We consider the actor to be the acting agent, i.e. the actor reasons and acts to calculate that the character it is playing would do.

We define three characters $c_1, c_2, c_3$ and an actor $a_1$ that plays $c_1$. $c_1$ has two opponents $c_2$ and $c_3$, and has a knowledge base for each of $c_1$'s opponents

**Figure 2.4.3** The character knowledge base structure




identical to $a_1$'s knowledge base in form, i.e. it is a causally structured BN based on the plot that DPGE generated, see Figure 2.4.3. $a_1$ has instantiated $c_1$'s opponents' knowledge base with what $a_1$ reasons that $c_2$ and $c_3$ would know.

Furthermore, $a_1$ also requires a mental image of what $a_1$ reasons that $c_2$ and $c_3$ would think that $c_2$ and $c_3$'s opponents would know. Therefore, $a_1$ has BNs based on the plot and instantiated with what $a_1$ reasons that $c_2$ thinks that $c_1$ and $c_3$ thinks. Similarly, $a_1$ has BNs based on the plot and instantiated with what $a_1$ reasons that $c_3$ thinks that $c_1$ and $c_2$ thinks.

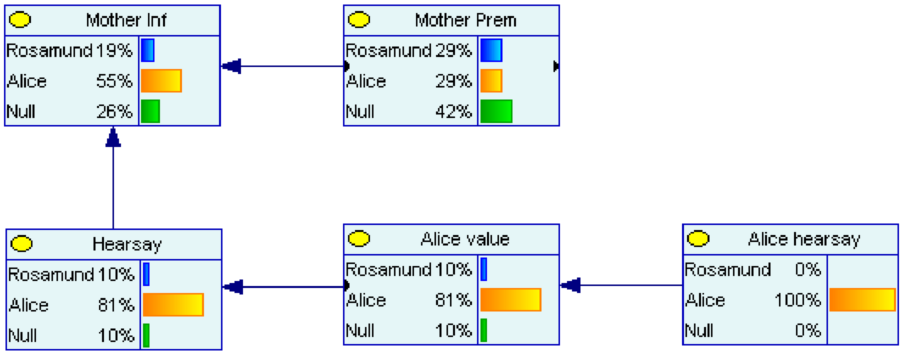
Therefore, $a_1$ has 7 BNs based on the initial plot and $a_1$ updates all 7 with any relevant new knowledge that $a_1$ learns.

We divide each of $a_1$ BNs into smaller subnets that contain two or more vertices describing the state of the distinct item of knowledge. For example, in Figure 2.4.4, a subnet contains the state of knowing who is the mother of another character. Each of the vertices has three states, two possible female names of other characters in this game, and null. The null represents the possibility that none of the other characters is the victim's mother.

The top right vertex, called "Mother prem" is connected to other vertices that have "prem" postfix in the net. Each of these knowledge piece subnets has precisely one such vertex that corresponds to the DPGE plot generated at the start. Some premise vertices are instantiated at the start of the game to represent what each actor knows from the DPGE generated plot.

 58

**Figure 2.4.4** Actor's knowledge subnet, displayed in the GeNIe UI [DSL07a]




The top left vertex "Mother inf" vertex is an inference of known facts from the "mother prem" vertex and inferred knowledge based on what other characters have told $c_1$. This vertex represents the inferred knowledge of both actual data from the premise vertex and information that $a_1$ has been told by an opponent, in this case, $c_2$ whom's character name is Alice.

The top left vertex "Mother inf" vertex is an inference of known facts from the "mother prem" vertex and inferred knowledge based on what other characters have told $c_1$. This vertex represents the inferred knowledge of both actual data from the premise vertex and information that $a_1$ has been told by an opponent, in this case, $c_2$, who's character name is Alice.

The "Alice hearsay" vertex shows what $c_2$ claimed to be the mother. The "Alice value" vertex shows how much value $c_1$ places on Alice's claims. The "value" vertices are influenced by the $c_1$'s gullibility and by how reliable $c_1$ believes $c_1$'s opponent is. Similarly, we add if $c_1$ talks to other characters and gives information on who the mother is. Thus $c_1$ can receive conflicting evidence from other characters and evaluate each hearsay given the opponent's reliability. The hearsay vertex aggregates all hearsay, and the inference vertex aggregates the premises and hearsay vertex. If the actor receives concrete evidence, then the "Mother prem" vertex will be instantiated to reflect that.

#### 2.4.2.2 Generating sentences and replies

We generate the sentences from the knowledge vertices. We use seed from another actor, player, event or interaction in the game. First, we find a knowledge vertex matching the seed. For instance, the *Mother Prem* vertex in Figure 2.4.4 would match the seed mother. Then all the parents and children of the *mother prem* vertex are retrieved. The premises connect to the mother prem vertex. These vertices are "relevant" because they either match the seed or have causal relations to a vertex that matches the seed.

Each state in a relevant knowledge vertex is used to create two sentences, one for the affirmative and one in denial. For example the *Mother Prem* vertex in Figure 2.4.4 would generate 6 possible sentences, namely: “Rosamund is the victim’s mother”, “Rosamund is not the victim’s mother”, “Alice is the victim’s mother”, “Alice is not the victim’s mother”, “none of the suspects is the victim’s mother” and finally “one of the suspects is the victim’s mother”. Knowledge vertices are also combined to create sentences from more than one knowledge vertex. For example, the following sentence contains three knowledge vertices: “Rosamund is a tall, blond female”.

We generate the replies in the same way as the sentences except that each of the knowledge vertices used to generate sentences is a seed for generating replies.

When an actor wants to say something she will:

1. Generate possible sentences and possible replies.
2. Calculate the optimal reply that she expects for each of her sentences.
3. Calculate her set of optimal sentences given the set of optimal replies.
4. Pick one sentence in the set of optimal sentences to speak.
5. Update her knowledge base with what she decided to say. If the opponent is an actor then the opponent will also update its knowledge base.

#### 2.4.2.3 Sentences and replies

Each time the actor wants to say something, it needs to generate a set of possible sentences. We generate sentences from the actors’ prior knowledge and their goals. It is also necessary to generate all the opponents’ possible replies to evaluate the effect of her intended sentence and see details in Section 2.4.2.2

The sentence and reply subnets contain the sentences that the active character could say and the replies she would expect from her opponent respective.

It is necessary to calculate the levels of contradiction and risk for each sentence and reply to get an evaluation of the over all risk of each sentence. First it is good to get a detailed description of what the contradiction and risk vertices measure and then it is possible to put it in a more general context by an example.

- *contradiction* is the degree of difference between the character’s statements and its theory of the opponent’s mind. For example, if John thinks that Linda knows that he broke a vase, John will measure a high contradiction when considering an opposing statement.

  To calculate the contradiction vertex of a sentence or reply it is necessary to compare the knowledge $k^{s,r}$ associated with each sentence $s$ or reply $r$ with each of the respective opponent's knowledge denoted $k^o$. Such that $k_1^{s,r}$ is compared to opponent's $k_1^o$.

Contradiction is caused by inequalities between the values of $k_1^{s,r}$ and $k_1^o$

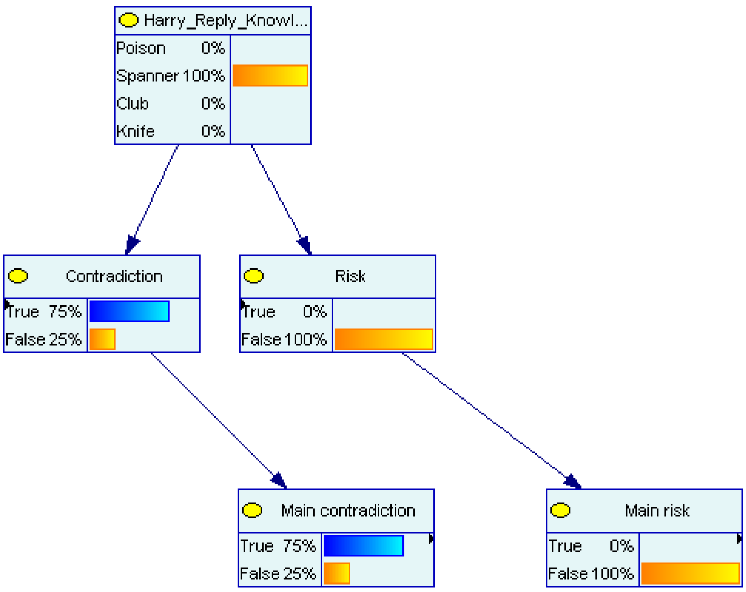
$$k_1^o = True \neq False = k_1^{s,r}$$

We argue that there is only a degree of contradiction since it is a measure of the character's belief of its opponents' state of mind, i.e. the character is reasoning under uncertainty. Moreover, it is reasonable to assume that there are degrees of contradiction due to other factors such as knowledge type. For example, if someone tells us he arrived at three o'clock, but we know that he arrived 15 minutes earlier, we do not necessarily consider it a contradiction, or at least less of a contradiction than if the same person claimed that he never arrived. Such effects would influence the contradiction vertex.

- *Risk* is when a sentence or a reply contradicts the knowledge of the actor. The estimated risk of a sentence follows the same process as the contradiction.

  The central contradiction and risk vertices aggregate values from all their parent contradiction and risk vertices. Having primary contradiction and risk vertices as intermediate steps is necessary to simplify the calculations.

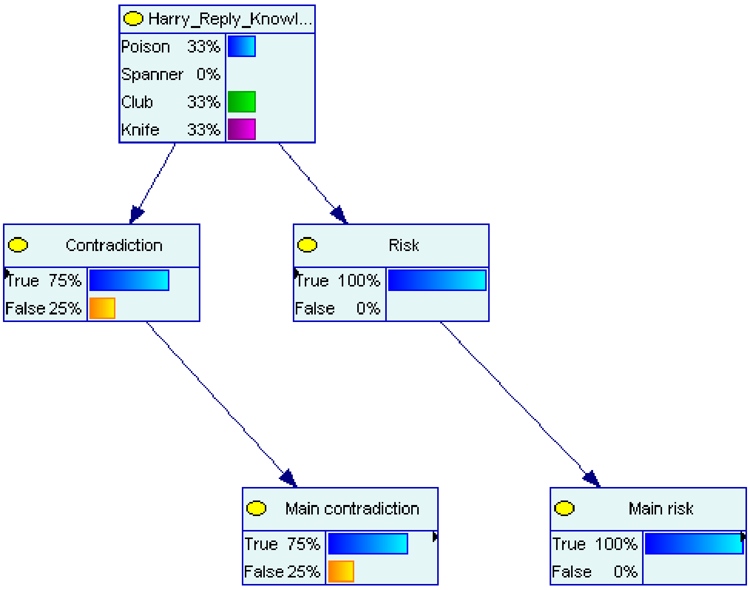
**Figure 2.4.5** Reasoning; is spanner true, displayed in the GeNIe UI [DSL07a]



For example, see Figure 2.4.5 and Figure 2.4.6 that show two possible replies: reply A "the murder weapon is spanner" and reply B "the murder weapon is not spanner". In this example, Harry is evaluating what the opponent is likely to reply. Harry believes that his opponent is sure that the murder weapon is a spanner and that the opponent believes that Harry is clueless about what the murder weapon is such that Harry has an equal value set on all states in his respective knowledge subnet.

The top vertex in Figure 2.4.5 has its probability for state spanner set to 1. and all other states set to 0. Since Harry beliefs that his opponent believes that Harry is clueless; Harry will assume that the opponent realizes that claiming that the murder weapon is spanner will contradict Harry's knowledge by 75%. On the other hand, Harry assumes that the opponent assumes no risk involved in claiming that the murder weapon is a spanner because the opponent is sure that the murder weapon is a spanner.

**Figure 2.4.6** Reasoning; is spanner false, displayed in the GeNIe UI [DSL07a]




In Figure 2.4.6 Harry evaluates possible replies to the effect that the murder weapon is not a spanner; so the probability of spanner is set to 0, albeit, all other states have equal probabilities. Just as in Figure 2.4.5 the contradiction is 0.75 because there is a contradiction of 75% essentially. The risk is very high because Harry assumes that the opponent will think that he, the opponent, is lying and will find it highly risky.

#### 2.4.2.4 RDE

The RDE is the *decision mechanism* of the actor to find a rational sentence.

We have a two-player, non-cooperative, sequential game in game-theoretic terms—one player against one opponent where the player can choose from n sentences (decisions). The player knows his type $\theta_i$, but not its opponents type $\theta_{-i}$. From exogenous data, the player knows all the opponent's possible types and all payoff functions contingent to each type. The player knows all possible replies the opponent can say, the opponent's payoff function, and an estimate of the opponent's type.

The game is sequential; the player speaks, and the opponent replies.

The type does not refer to character types such as suspect or murderer. Instead, it refers to the status of the actor's knowledge base. An actor does bot know of an affair is of one type, but when the same actor learns about an affair, he becomes another type. In the worst case in a brute force algorithm, all possible types of a player would mean a cross-product of all the vertices' states in the actor's Bayesian net.

A MAID [KM03], introduced in Section 2.3.2.4 is an extended Bayesian net for finding equilibrium for the agents' decision problems. Koller and Milch refer to our problem domain as a "single-agent decision problem" [KM03], and the players take turns.

For each run of the engine, it needs to set up the Bayesian game. First, it generates sentences and all possible replies to each of the sentences from the knowledge base. We describe the knowledge base in the next section, followed by a section describing how to generate the sentences and replies.

The engine finds an equilibrium, a set of optimal sentences using MAIDs, and the actor picks one to say, see Section 2.4.2.5. The engine also adds any knowledge gained to the knowledge base.

The actor receives information from other actors and players, evaluate it and add it to its knowledge base. It allows unstructured emergent behaviour from the interactions between the player and actors.

The actors make no distinction between another actor and a human player, so if a human player takes the part of a suspect or murderer defined by the plot engine, the actors will interact with the human player just as if the human player were another actor.

The outcomes of the RDE are measured in Game-Theoretic terms, not whether the dialogues appear natural to humans. There is no structure applied to the engine itself to present it as, for example, lines in a drama or a fluent conversation with humans. To structure the rational dialogue output, we need another engine such as the DED.

#### 2.4.2.5 Finding equilibrium

The actors want to find an equilibrium to determine a set of optimal sentences from those generated from the seed. It is important to realise that the game theoretic "game" that the actors face is sequential, e.g. the actor makes a move and then his opponent makes a move. They do not make their moves in parallel which simplifies the computation of equilibrium significantly.

First the engine sets up a game theoretic game using Multi Agent Influence Diagrams (MAIDs) [KM03].

The MAIDs offer an algorithm to find equilibrium in incomplete information games for single agent decision problems in linear time. The MAIDs map the decision vertices of the Bayesian net with a "relevance graph" to discover in what order the decisions should be evaluated. There are two decisions that need to be evaluated by the actor: The assumed reply of the opponent and the sentence that the actor should speak with respect to the assumed reply.

Since the set of optimal sentences is dependent on their replies the first step is to calculate the set of optimal replies to each of the sentences. The second step is to determine the set of optimal sentences with respect to the replies that each of them is liable to receive.

This means that the engine generates all the possible strategies (sentences) that the actor can play, $s_i'(\theta_i)$, and all the strategies (replies) that the opponent can play contingent on each of their possible types $s_{-i}(\theta_{-i})$. This is done by generating the sentences and replies as previously described.

What the actor will say is dependent upon what he expects as a reply. So we have two decision variables in a MAID. One for the optimal sentences $D_s$ and another for the optimal replies $D_r$. Decision variable $D_s$ is dependent on decision variable $D_r$. This means that we can draw a relevance graph such that $D_s \to D_r$. This means that there is need for two MAIDs to find an equilibrium. One induced by the replies $\mathcal{M}_{[\sigma_{D_r}]}$ and the other induced by the sentences $\mathcal{M}_{[\sigma_{D_s}]}$. There are then two MAIDs: $\mathcal{M}_{[\sigma_{D_r}]}$ where $D_r$ is a chance variable and $\mathcal{M}_{[\sigma_{D_s}]}$ where $D_s$ is a chance variable. When these two MAIDs have been constructed the algorithm evaluates the two MAIDs by iterating through them in reverse order. This means that $\mathcal{M}_{[\sigma_{D_r}]}$ is calculated first and $\mathcal{M}_{[\sigma_{D_s}]}$ second induced with the results from $D_r$. The decision reached in $\mathcal{M}_{[\sigma_{D_s}]}$ is then an optimal decision. This is equivalent to finding a Nash equilibrium.

These strategies are then used to find the weighted payoff as detailed for the Bayesian equilibrium [FT91],

$$p(\theta_{-i}|\theta_i)u_i(s_i', s_{-i}(\theta_{-i}), (\theta_i, \theta_{-i}))$$

Next the engine sums up the payoff for each of the player's strategies and from the sum collects the set of strategies that carries the maximum value.

Finally the actor can simply pick some random sentence from the set of optimal sentences to speak. The engine adds any knowledge gained to the actor's knowledge base as described in Section 2.4.2.1.

In order to better clarify how the engine evaluates sentences I will take as an example an actor named Horace that is talking to an opponent named Linda. Horace has just been asked whether he owed the victim money. First the engine generates a set of possible sentences for Horace to speak. Horace could both say "I'm in debt" or "I'm not in debt" and he could also say "I'm rich" or "I'm poor" since that has a logical connection to whether someone is indebted or not. There could of course be other options but these will do as an example. The engine also creates all possible replies to each of the possible sentences.

If Horace was indebted to the victim then he may want to lie as admitting to having a motive will place him under suspicion. Now lying has risks; if Linda catches Horace in a lie then Horace will appear to be even more suspicious than if he tells the truth. Horace may also have high moral standards and not care to lie.

Let us say that Horace was indebted to the victim, then admitting to it will carry a penalty due to increased suspicion. Denying it will carry a penalty due to risk of being caught lying and possibly because of moral standards. This would mean that admitting to being indebted is likely to get a higher payoff, especially if the engine calculates that the optimal response to denying the debt would indicate that Linda knows that Horace is lying. Horace can also state that he is poor. That may have the highest payoff. Being poor is not a motive for murder, it is not contradicting what Horace knows is true and unless he believes that Linda may claim that he is lying then claiming to be poor is likely to carry the highest payoff in this example. If Horace is not indebted to the victim then either claiming to be rich or denying the debt is likely to carry the highest payoff.

When Horace has chosen a sentence to speak then his opponent's knowledge base is updated with what Horace just said as described in Section 2.4.2.1. Horace's knowledge base is also updated so that Horace will remember what he told Linda.

If Linda is an actor then Linda's engine would start by adding the knowledge acquired from Horace's sentence to Linda's knowledge base. Then from the newly updated knowledge base the engine will generate all possible sentences for Linda to speak using Horace's sentence as a seed. Then Linda's engine draws up a new game theoretic game and the engine finds an equilibrium for Linda just as Horace's engine found an equilibrium for him. It is important to understand from this example that:

1. Finding one sentence to speak is one game theoretic game.
2. The goals of the actors are subjective such as not to lie or become suspicious.
3. There are no explicit goals, such as solve the mystery.
4. Characteristics, feelings, knowledge, beliefs about the opponent's reactions etc. influence the actor payoff.
5. The actors add to their knowledge base and the next game theoretic game reads from an updated knowledge base.

6. The actor does not care whether he is talking to a human or another actor, it is not a factor in the engine's calculations.

#### 2.4.2.6 Output of the RDE

The actors are able to calculate an optimal sentence using GT and they are able to participate in a dialogue, see Example 2.4.1.

**Example 2.4.1 (Calculating optimal sentence)**

*Rosamund and Alice are talking together and Rosamund got the seed wedlock which is a possible motive for murder. The conversation drifts into stating their actual relations with the victim and ends when Rosamund has nothing to add to the conversation.*

1. *Rosamund says: Horace did not hold me in wedlock.*
2. *Alice says: Horace did not hold me in wedlock.*
3. *Rosamund says: I'm not Horace's wife.*
4. *Alice says: I'm not Horace's wife.*
5. *Rosamund says: I'm Horace's parent.*
6. *Alice says: Horace did not have an affair with a male.*
7. *Rosamund says: I'm Horace's mother.*
8. *Alice says: Horace was having an affair with me.*
9. *Rosamund can't think of anything to say!*

## 2.5 Summary

In the Narratology Section 2.2 we introduced the work and theories of several structuralists and experts in drama-theory and narratology. We discussed how narrative and drama are an integral part of games and PP-RPG are effectively real-life play and improvisation, which has been difficult to transfer into CB-RPG and retain the former's free nature.

In the latter half, we introduced complex technology, BNs, GT and the combination of the two into the MAIDs. We described the DPGE based on causally structured BNs and generates plots dynamically from static input. Followed by our RDE that uses the MAIDs. The RDE calculates a rational sentence for an actor to speak. It uses a knowledge base, of causally structured BNs, to generate sentences and replies and calculates an equilibrium according to GT. The actors evaluate the set of possible sentences by what they think and what they expect their opponents to think and reply.

Moreover, the actors can adapt to what they hear from their opponents. They do this by adding the hearsay nodes to their opponent's last spoken sentence. As shown in Example 2.4.1, the sentences generated are far from what humans would expect in a dialogue. The RDE does not generate sentences that humans would find 'interesting', 'natural' and 'fully coherent' as we see it.

Initially, the engine was too slow to respond about 50% took more than 1 minute to compute. It helped that most of the time, there were not that many sentences generated. In the worst case, belief updating algorithms for BNs are NP-hard [Coo90]. There are several approximation algorithms available, see Table 2.5.1. All of these algorithms have an NP-hard worst-case [DL93]. BNs' complexity grows with the number of vertices and connections between vertices.

The network used for the actors' current implementation has over 3000 vertices, which is not a very big network for an intelligent virtual agent. An average agent in an interactive drama could easily have a few hundreds of thousand vertices.

However, even this small BN takes several seconds to update with basic BN reasoning algorithms [Ari07a, Ari07b], which is not sufficiently fast for real-time applications where the player expects an answer from the NPC in about one or two seconds. Not only is it not fast enough, but it also leaves no room for all the other processes, such as graphics and animation that are highly demanding on resources in any high graphics computer game. All the algorithms available in the Smile library[DSL07b] are shown in Table 2.5.1. We tried them all, but none of them reduced the computation time significantly. All the algorithms in Table 2.5.1 are attempting to update the whole network.

The DED that we introduce in this thesis resolves some of these issues. First, it addresses the previous problems of lack of coherent behaviour

Table 2.5.1 The algorithms available to compute the values of a BN

| | |
|---|---|
| LAURITZEN | Probability Propagation [HD96] |
| HENRION | Probabilistic Logic Sampling [Hen88] |
| PEARL | Fusion, Propagation, and Structuring [Pea86] |
| LSAMPLING | Simulation with likelihood weighting [SP90] |
| SELFIMPORTANCE | Simulation with self importance [SP90] |
| BACKSAMPLING | Backward Simulation [FF94] |
| AISSAMPLING | Adaptive Importance Sampling Algorithm [CD00] |
| EPISSAMPLING | Evidence Pre-propagation Importance Sampling Algorithm [YD03] |

by using goal-driven behaviour and developing algorithms that utilise the causal structure of the actors' knowledge base. It adds generic methods to select actions that increase the characters' position and whether they are causally related to specific character goals and drama goals. We define these goals in schemas, which are motif like structures similar to G. Polti's situations [Pol21]. We also apply a more sophisticated evaluation of knowledge that the actor receives before adding it to their knowledge base. We address the performance issues by applying Relevance Reasoning [LD97] and OOBNs[KP97].

# 3

# Related work

## 3.1 Introduction

First, we establish drama metrics for empirical evaluation of related systems; to effectively compare them to the DED, see Section 3.2.

In Section 3.3 we review related systems and measure them against the drama metrics.

Finally, we present and discuss the result of our measurements, see Section 3.4.

## 3.2 Drama metrics

We build our Definition 1.1.1 of emergent interactive drama on:

1. Our published *Critical Review of Interactive Drama Systems* [ABK09]
2. H. Barber's Compellingness criteria is defined to include all required features of an interactive narrative. [Bar08]
3. Our requirements for believable NPCs

Compellingness, H. Barber says:

> *" is defined to include all required features of an interactive narrative. The compellingness criteria must be satisfied by an interactive narrative system. ... interestingness, immersion, scalability, domain independence, agency and replayability. ... An interactive narrative is a game world in which the user-controlled*

 69

> *character(s) can physically and mentally interact with ideally perceived total freedom while experiencing a dramatically interesting story which is fundamentally different on nearly every play - dependent on the user's actions.*
> ”
>
> – [Bar08, p. 19]

Our review's related systems have their definitions of interactive drama, agency, believability, ideal narrative structure; their goals often differ from our goals and require different emphasise.

From H. Barber's definition, we can define metrics to quantify how well a system supports interactive drama. We assign an inverse numerical value to a subjective categorisation of how well the system supports a feature to avoid inflating the values, see Definition 3.2.1. A 3D game environment or VR accommodates the highest degree of freedom versus simple or text-based environments. We measure separately whether the systems offer means of interacting with objects and NPCs.

**Definition 3.2.1 (Drama systems' metrics)**

| ***level*** | ***(GT) Graphics' Type*** | ***(OI) Objects' interaction*** | ***(SI) Social interaction*** |
|---|---|---|---|
| *1* | *3D game environment or VR* | *Yes* | *Yes* |
| *2* | *Simple graphics* | *Some* | *Some* |
| *3* | *Text-based* | *No* | *No* |
| *4* | *None* | *N/A* | *N/A* |

In Definition 3.2.2 we define a measure for how many fundamentally different narrative variations the system generates from a single setup. This doubles as a measurement of how interesting and replayable a system is since players will generally not replay a game that they find uninteresting or that does not offer any variation on subsequent plays.

**Definition 3.2.2 ((FD) Fundamental difference levels by order of magnitude)**

| ***level*** | ***O()*** | ***Descript.*** | ***Explanation*** |
|---|---|---|---|
| *1* | $O(\infty)$ | *Infinite* | *Intractable; theoretically impossible to author all variations* |
| *2* | *O(10)* | $> 10$ | *Challenging; theoretically possible to author all variation* |
| *3* | *O(1)* | $<= 10$ | *Feasible* |

Our narrative metrics, see Definition 3.2.3, measure:

- The use of Narratology to structure the interactive drama.
- Agency of the players and their influence on the plot.
- Autonomy, drama awareness and influence of the NPCs on the narrative.

**Definition 3.2.3 (Narrative metrics )**

| | ***Category*** | *1* | *2* | *3* | *4* |
|---|---|---|---|---|---|
| *(PS)* | *Plot structure* | *Abstract* | *Stochastic* | *Discrete* | *linear or None* |
| *(ST)* | *Structuralism* | *Yes* | *Some* | *Minimal* | *None* |
| *(DT)* | *Drama theory* | *Yes* | *Some* | *Minimal* | *None* |
| *(PR)* | *Player role* | *Protagonist* | *Secondary* | *Minimal* | *None* |
| *(PI)* | *Player influence* | *Significant* | *Some* | *Minimal* | *None* |
| *(AN)* | *Autonomous NPCs* | *Yes* | *Some* | *Minimal* | *None* |
| *(NI)* | *NPCs influence* | *Yes* | *Some* | *Minimal* | *None* |

In Definition 3.2.4 we define metrics that measure the believability of NPCs.

**Definition 3.2.4 (Agent metrics )**

| | ***Category*** | *1* | *2* |
|---|---|---|---|
| *(LR)* | *Logical reasoning* | *Yes* | *No* |
| *(SI)* | *Stochastic inference* | *Yes* | *No* |
| *(Ra)* | *Rational, GT* | *Yes* | *No* |
| *(TM)* | *Theory of Mind* | *Yes* | *No* |
| *(Em)* | *Emotions* | *Yes* | *No* |
| *(CT)* | *Character traits* | *Yes* | *No* |
| *(DA)* | *Drama awareness* | *Yes* | *No* |
| *(Du)* | *Duality* | *Yes* | *No* |
| *(KB)* | *Knowledge base* | *Yes* | *No* |
| *(TE)* | *Truth evaluation* | *Yes* | *No* |
| *(Le)* | *Learn* | *Yes* | *No* |

In Section 3.3 we add a table by each system with its values across all the metrics. These figures are overestimates of the potential, as exact numbers are not known.

## 3.3 Related system

### 3.3.1 BARDS [PCLC07, PC07]

Has a Heuristic Search Planner (HSP) with RTA* as its underlying search algorithm to plan emotional development in the characters rather than actions [PCLC07, PC07]. A user can interact with the story by natural language so that user comments can cause an emotional reaction with the characters. For example, to remind an adulterous woman of her children to induce guilt.

The emotional plan, actions and feelings are developed according to what the character is feeling and alters the storyline.

The interaction has the feel of movie theatre audience shouting at their hero: “beware behind you" when the hero is in danger. They emphasise believability through emotions displayed by their characters depending on the backstory of the characters. However, their player does not play a character in the story.

**Table 3.3.1** BARDS metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 1 | 3 | 2 | 2 | 3 | 1 | 1 | 3 | 3 | 3 | 3 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 2 |

### 3.3.2 The Mimesis system [YR03, RSY03, YRB+04]

The Mimesis system uses planning to reconstruct a pre-authored narrative where the player plays the protagonist. They designed the system to allow the player to choose from a wide range of actions, but at the same time, the system goes to great lengths to hinder the player from affecting the intended narration. The system can use the following story mediations:

1. Accommodation, the system re-plans to allow for this action.
2. Intervention, an action by the player is unsuccessful, such as shooting a character misses or jamming the gun.

The system first tries to accommodate the action by re-planning based on changes resulting from the action. When the system receives a plan request, it draws a DAG of the intended narrative's actions to play out correctly.

Generating a DAG in this way is feasible for short stories or a limited range of actions. It is though infeasible for larger stories due to the state explosion of planning in general. For this solution to work, it is imperative to keep the action plans small. It can help to divide the plans into smaller problems or hierarchical structure.

An unforeseen action that would disrupt the intended narrative is intervened by, for example, the player missing or the gun jamming when trying to shoot the main character. The system does not allow any fundamentally different stories to emerge; they design the mediations to block any diversion.

**Table 3.3.2** Mimesis metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 1 | 3 | 3 | 3 | 1 | 3 | 1 | 4 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.3 IN-TALE [RS06a, RS06b, RS06c]

IN-TALE is the same viewpoint as mimesis. They use plot points and planning, and their aim is for the user to fallow a specific narrative path.

The player finds himself in a marketplace where a merchant called Mohammed plants a bomb in a stall owned by merchant Saleh. When the bomb goes off the player needs to apply their leadership skills to get the marketplace back in order and hopefully charge the correct merchant.

The main emphasis of IN-TALE is around the planting of the bomb. The player can stop the bomb from going off in the first instance. If the player is successful in stopping the bomb, the chaos does not ensue. The leadership training is the purpose of the game and, therefore, the bomb must go off.

For this purpose, they developed Narrative Directive Behaviours (NDB). The director can send the actors NDBs, making the actors drop their current goal gently and believably and start to follow the NDB goal, which alters the drama so that the bomb goes off. For example, the merchant may rearm a disarmed bomb.

**Table 3.3.3** IN-TALE metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 1 | 1 | 2 | 2 | 3 | 1 | 1 | 1 | 3 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.4 Interactive Drama Architecture (IDA) [MLA+04, ML04, Mag05,

Mag06]

The IDA system has an author, director agent, synthetic characters and a player. They built their testbed, Haunt 2, in Unreal Tournament where the player as a ghost needs to find his murderer and manipulate other characters to find the body and the murderer. Having the player be a ghost explains the players' lack of freedom and significantly reduces the system's complexity. IDA is in collaboration with the Soar Games Group and draws on lessons from Mimesis.

An author or a group of authors are required to author the story. The authoring includes any domain-dependent functions of the director, the environment and art content, and all characters' behaviours. The characters are semi-autonomous in that they can occupy themselves when they have no instructions from the director. They, for example, light a fire and eat, drink and chat. The agents mainly take commands from the director. The director commands take priority over all other goals. The commands can be high level such as "explore" and also precise: “perform dialogue #131 with John in the library and then run away to another room."[Mag06]

The story itself is drawn up as plot points in a partially ordered graph using STRIPS with pre and postconditions. It faces the same difficulty with intractability as the story grows and has more characters and actions.

IDA also uses story modifiers as Mimesis to maintain the narrative. IDA philosophy is slightly different as they recognise that too much direct interaction compromises believability; for this reason, they introduce director actions that shift or modify the plot to accommodate player actions.

- Deniers, permanently or temporarily make certain plot points inaccessible.
- Causers, the system initiates a plot point.
- Creations, new things appear in-game to replace destroyed items.
- Shifters, plot points are moved around to aid the player.
- Hints, some noise from a room or an NPC says something.

**Table 3.3.4** IDA metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 3 | 2 | 3 | 3 | 2 | 1 | 1 | 4 | 3 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

**3.3.5 Thespian** [SMP05a, SMP05b, SMP06b, SMP06a, SMP07, SMP08c, SMP08a, SMP08b, SMP09a, SMP09b, SMP10, SM10]

The Thespian system aims to create Interactive Pedagogical Dramas (IPD) based on authored scripts. The aim is for the agents to play out the script when they have the opportunity to do so. The agents are given the key points of the script as character goals. Thespian agents try to satisfy their goals, and if the player plays it right, the player can experience the complete intended drama. Thespian is by this realising Pedagogical goals by forcing the player by trial and error to learn the correct way through the drama.

Thespian's example domain is to teach users a foreign language and cultural awareness. The user takes on the role of a male army sergeant assigned to conduct a civil affairs mission in a foreign (e.g., Lebanese, Iraqi) town. The user navigates in a virtual world and interacts with virtual characters using spoken Arabic and gestures. Thespian agents have a look-ahead search in a decision-theoretic framework to determine how best to achieve their goals. The look-ahead search is when the agents speculate based on their belief of how other agents and the player react to their actions. Their goals become a critical determinant of their behaviour, and the agents try to respond to unexpected user interaction in ways consistent with their motivations.

While Thespian has goal-oriented agents and these agents are autonomous to some degree; it is essentially following a script, and the aim is to get the player to learn the correct behaviour to realise the scripted drama. The purpose is to have the player go through the game's educational lessons, i.e. Arabic manners.

**Table 3.3.5** Thespian metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 1 | 3 | 1 | 3 | 2 | 1 | 3 | 1 | 4 | 2 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 1 | 2 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 2 |

### 3.3.6 IDtension [Szi01, Szi02, SMR03, Szi03, Szi04, SR04, SM05, Szi05, Szi07a, Szi07b, Szi08, Szi10, SRP12, SAR12]

IDtension bases its approach on Narratology such as Propp's functions, Bremond's process, Greimas actant's model and Todorov's transformations on which they design their theory: "we explicitly design our own theory of narrative and drama, both influenced by existing theories and the specific needs of a computer simulation." [Szi03, p. 2].

They divide their interactive narrative into three layers:

1. The discourse, layer which carries the message or theme of the story.
2. The story, layer which is a succession of events and character actions, based on several structuralistic rules and narrative sequences.

3. Perception, a discourse on how to perceive the narrative during reading/viewing/listening.

The IDtension system is authored by defining and scripting a set of causally ordered tasks to complete a specific goal. There are several such causal pathways to complete each goal. IDtension models the narrative by a user model that uses the following criteria as seen in [Szi03, p. 6]:

- Ethical consistency: The action is consistent with the same character's previous actions concerning the system of values.
- Motivational consistency: The action is consistent with the goals of the character.
- Relevance: The action is relevant to player actions.
- Cognitive load: The action opens or closes a narrative process, depending on the current number of opened processes and the desired number of opened processes (high at the beginning, null at the end).
- Characterization: The action helps the user to understand the characters' features.
- Conflict: The action either exhibits some conflict directly. For example, an incentive that conflicts with the inciting character's values, or the action pushes the user towards a conflicting task. For example, block a non-conflicting task, if a conflicting task exists.

The idea has an absolute novelty in CS, offering a range of actions that lead rather than block. It has though a considerable authoring bottle- neck as the author needs to design all the possible tasks, and the author is unlikely to anticipate all the player's expectations. The author cannot possibly fathom which tasks to include. An author who attempted to author a story for IDtension experienced authoring problems due to a high degree of abstraction used, alienating the author from authoring the character's personality. This method also poses the same intractable planning problems as in Mimesis and IDA if the intention is to have predefined goals reached with each story.

**Table 3.3.6** IDtension metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 3 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.7 Façade [Mat02, MS05a]

The player finds herself in a dreadful situation where her old friends Trip and Grace, who has just invited her over, are clearly on the brink of a marital breakdown [Mat02, MS05a].

The player can move around the scene and point to things, and the characters react. If a player points at an object, the engine instructs the characters how to react. In the words of M. Mateas, the instructions are beats, which are:

> *" In the Façade architecture, the unit of plot/character integration is the dramatic beat. In the theory of dramatic writing, beats are the smallest unit of dramatic action, consisting of a short dialogue exchange or small amount of physical action. "*
>
> – [Mat02, p. ii]

The player can also talk to Trip and Grace via natural language processing. Trip and Grace do not understand what the player says, according to M. Mateas & A. Stern:

> *" Grace and Trip interpret all of the player's discourse acts in terms of a zero-sum affinity game that determines whose side Trip and Grace currently believe the player to be on. "*
>
> – [MS05a, p. 4]

Trip and Grace are shallow agents that are incapable of any awareness of the unfolding drama. If the player does nothing, Trip and Grace talk to each other, going through one beat after another.

As the drama progresses, Trip and Grace start to argue and eventually split up if the player does not talk them out of it. There is a lack of an inciting event in the drama. Nothing happens at the start of the drama to get the player interested in the marital dispute outcome. The player is left guessing what their role is.

They say that Façade is the first wholly implemented interactive drama, which is valid for the academic field of interactive drama. Many commercial RPGs, where the player can bargain, attack, kill, steal, flirt, manipulate, form friendships, accept/reject quests offer a more significant amount of interactivity than Façade. For example, Deus Ex [Inc00] (released in June 2000) and RPG Morrowind, (released in May 2002) and have multiple possible endings. These games are certainly dramas and are considerably longer and more intensive than Façade. Therefore, it does not hold in a commercial sense that Façade is the first wholly implemented interactive drama.

Façade has a Drama Manager, beats, characters, story values, actions and natural language processing.

The beats are pre-authored with explicit actions for all roles that are strictly coordinated to allow for multi-agent coordination, in M. Mateas words.

> *" Roughly, a beat consists of an action/reaction pair between characters. For example, in the case where action is being carried by dialog, a beat could simply consist of one character speaking a line of dialog, and another character reacting. Generally speaking, in the interest of maintaining economy and intensity, a beat should not last longer than a few actions or lines of dialog. "*
>
> – [Mat02, p.45]

Additionally, all higher-level goals and behaviours that drive a character are in the beats rather than the character. The character retains some autonomy when it comes to base level goals and actions such as facial expressions or personality moves, as M. Mateas says:

> *" As architectural entities, beats organize both the procedural knowledge to accomplish the beat's dramatic action, and the declarative knowledge to sequence the beat in an evolving plot. Instead of conceiving of the characters as strongly autonomous entities that coordinate to accomplish dramatic action through purely local decision-making, characters are instead weakly autonomous. "*
>
> – [Mat02, p. ii]

According to M. Mateas & A. Stern:

> *" Façade required 3 person years of just authoring ... only 27 beats were created in the end. "*
>
> – [MS05a, p. 6]

The result is a 20-25 minute long interactive drama that you can play through 3-4 times and get some novelty.

Façade has an algorithm constantly selecting which beat to perform next based on the level of tension and affinity as M. Mateas says:

> *" Move Wedding Picture (artist/ advertising). Tension 2, Trip Affinity. At this point, two tension 1 story topic beats have completed, so tension 2 beats become sequenceable (except for Decorating and Ask About Drinks, which have already been performed at tension 1). At eight beats into tension 1, the desired tension, as specified by the tension arc, is 2. Tension 2 beats get a higher score against the story arc, and thus have a higher probability of being selected. (though the probability of selecting a tension 1 beat is not zero). Additionally, because the player has maintained affinity with Trip for three beats in a row, the singlet Move Wedding Picture becomes sequenceable, and is in a higher-priority tier*

*(equivalent to a discourse act reference to the beat topic) because of its static priority. Since this is the only beat in the higher-priority tier, the beat manager sequences it. The player takes Grace's side, causing the affinity to switch to Grace. This beat also causes the tension to move to 2. "*

– [Mat02, p.171]

**Table 3.3.7** Façade metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 2 | 2 | 2 | 3 | 1 | 4 | 3 | 1 | 3 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.8 U-Director [ML06]

U-Director uses HTN planning and dynamic decision networks to implement a medical mystery story on a secluded island [ML06]. The story is pre-authored and follows a strict plot. The idea is very similar to IDA except that they use a Bayesian inference mechanism instead of planning when deciding how to manipulate the player into following the plot.

The director agent is constantly evaluating user actions and comparing them with the plot. The director attempts to engage the player in the plot by giving hints that steer the player in the right direction. If the hints are insufficient, the director uses more force. For example, have a character take the initiative rather than wait for the player to ask. The director uses extended BNs to evaluate what directive action to take based on maximising expected narrative utility. The director primarily progresses the narrative and tries to involve the player.

**Table 3.3.8** U-Director metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 2 | 2 | 3 | 3 | 1 | 4 | 1 | 4 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.9 FAtiMA [PDS+04, PDS+05, AFL+06b, AFL+06a, LADP05, FDA+07, VSPA07, LAD+07b, LAD07a, FPA07, ALT+08, FBAP08, LDAP08, LDAP09]

An anti-bullying interactive drama where the player, plays the same age confidant for a child experiencing bullying. The drama introduces the school, the children in the school and any other characters to provide a starting context. There are four roles in bullying situations the bully, the victim, the bully-victim, a child that is sometimes the victim and sometimes the bully, and the bystander. [PDS+05, p. 241] The player witnesses a bullying incident after meeting the victim in the library. The victim seeks advice from the player in how to deal with the bullying. The player is asked a few questions in dialogue form, as R. Aylett & Co says: "John can you think of a solution for Luke to stop bullying me?"[AFL+06a, p. 3]. They have a free-form text interface where the player can type what she wants. Behind the scenes is a template based map, in R. Aylett & Co words:

> *" the agent receives the utterance and converts it to a language action - one of the coping responses using a template-based language system "*
>
> – [LADP05, p. 48]

When the player has chosen a coping choice, the victim decides whether to go through with it. The victim does this according to its characteristics and emotions. For example, a timid victim will perhaps not choose to go through with an aggressive recommendation. The victim chooses an action based on the recommendation and faces another bullying scene. If the recommendation was excellent, and the victim successfully carried it out, then the bullying is resolved, if not then the above sequence repeats until the bullying is resolved [PDS+05, LADP05, p. 242,p. 44]. In the end, there is some educational message and in some cases, a questioner.

When the characters react, they use a set of emotional reaction rules based on appraisal values, including desirability and desirability for others, praiseworthiness. Preconditions match the rules with the current situation. The characters are also deliberate or goal-driven for which they use STRIPS-based partial-order, continuous planner. They evaluate the probability of success and the importance of the actions and whether they generate hope or fear. The action likely to generate the most potent emotion is chosen [AFL+06a, p. 4-5]. This architecture is called FearNot! Affective Mind Architecture (FAtiMA).

FAtiMA also applies a theory of mind [LA07, AL08].

1. Double appraisal: When the agent has chosen the action that would cause the strongest emotion, it feeds all the actions generated back into its appraisal system to determine which actions evoke the most vigorous emotional response.

2. Re-appraisal: feeds the actions into all the scenario characters' emotional systems to determine which action causes others' strongest emotional reaction.

A Story Facilitator(SF) sequences the episodes of the drama. The SF's episode plan is like a state machine, as R. Aylett & Co says:

> *" For an episode to be selected by the SF, it must have at least one of its preconditions satisfied. A precondition represents a set of tests on events, that when true indicate that this episode fits into the developing story. "*
>
> – [AFL+06a, p. 8]

The episodes are structured as follows [FDA+07, p. 5] and [ALT+08, p. 5]:

- Name: a unique name for the episode.
- Set: the set is the location in the virtual environment where the events of this episode will take place.
- Characters: the characters of the story, defined through a set of properties like their name, position on the set.
- Preconditions: a set of conditions that specify when is the episode eligible for selection.
- Goals: character goals for the agents.
- Triggers: a condition that executes a set of narrative actions.
- Finish conditions: a set of conditions similar to the preconditions to complete the game.
- Introduction: a set of narrative actions introducing the episode and characters, some introductory text.

Finally, FAtiMA is the only interactive drama system that we know of, which has gone through proper evaluation of whether it effectively reduces bullying. The results were promising, although more trials and development would be needed to evaluate long term affects and overall prevalence, in the words of M. Sapouna & Co:

> *" The close-response relationship within the intervention group indicates that those more actively engaged with the characters, rather than those who passively watched a greater number of episodes, were more likely to escape victimization, supporting our interpretation. We speculate that the interaction with FearNot!, at least temporarily, boosted victimized children's self-confidence in their ability to deal with bullying as they vicariously experienced successfully responding to bullying in the virtual world. "*
>
> – [SWV+09, p. 7]

**Table 3.3.9** FAtiMA metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 |

### 3.3.10 Gadin [BK07, Bar08]

Gadin is an emergent system for creating clichéd dramas as are commonly found in soaps. The system initially used STRIPS action planning for characters to generate dilemmas according to the characters personality and current interests. A drug addict will want to drug and may find itself in prison while a non-drug addict is far less likely to have such bleak prospects.

However, they found that the additional complexity of planning the dilemmas was an unnecessary overhead. Instead Gadin will periodically generate dilemmas to create conflicts in the unfolding drama. The possible types of dilemmas are:

- Betrayal: To betray their partner or close friend.
- Sacrifice: take damage for the good of a friend or partner.
- Greater Good: Do something good for many, including the players enemy.
- Take Down: Damage itself, but also damage its enemy even more.
- Favour: choose which friend or close partner is favoured.

The engine matches dilemmas to player actions and the current storyline in order to maintain a coherent story. It presents the characters with dilemmas according to their actions and place in the current drama for a realistic character-based decision. Gadin is an ingenious and genuinely novel approach to interactive drama. It has no pre-written script and creates near-endless soaps.

**Table 3.3.10** GADIN metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 2 |

### 3.3.11 NOLIST [BJJ+04]

In the non-linear interactive storytelling game engine (NOLIST) they utilise a BN to determine the culprit of a murder mystery; the BN is dynamically changing in response to the player's actions and observations. It is not preset but instead uses the player's moves and the logical inference of the net to determine the culprit. For example, finding a gun beside a body increases the probability that a gun is the murder weapon. The aim is to create a dynamic emergent storyline from player actions and choices. The engine creates the plot in the course of the game. Thus NOLIST creates the past in reaction to player interaction.

Their engine plots the progression of a player through a murder mystery. The game engine was not ready when the article was published, and there were no follow up publications.

> *" ... we envision a game engine that, whenever it has to choose an action, ensures that the game generated will have an ending, regardless of the players actions. It furthermore has to ensure that the possible continuations of the game will be interesting, and that two games will not be identical even if the player always performs the same actions. "*
>
> – [BJJ+04, p. 7]

**Table 3.3.11** NOLIST metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 1 | 2 | 1 | 1 | 3 | 2 | 1 | 1 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.12 Open Ended Proppian Interactive Adaptive Tale Engine (OPIATE) [FC03, FC04, Fai04]

The engine creates emergent interactive drama and is in large part based on Propp's Morphology [Pro68]. The game engine uses a story director who guides actors by giving them goals relevant to the story world's events. The drama emerges both from character interactions and events initiated by a director. The story director uses Proppian functions to generate an open-ended story world and chooses functions based planning that uses the current state of the story world to find the most reasonable goals to advance the story.

- Social simulation - the actors have an impressive gossiping system that indexes gossip by events and persons and ranks the gossip to structure what is select to gossip.
- Idle behaviours - patrolling or following others when they do not have a goal to chase.

- Targeted behaviour - the director assigns actors goals. The actors commit to one goal at a time. Goals are, for example, an object. The object can have a behaviour that the actor then executes, i.e. (goal = sword), the actor finds the sword picks it up and uses it to slash someone if that is the goal's action.
- Attitudes - actors develop attitudes about other actors and the player, both through direct contact and via the gossiping system.

The testbed for OPIATE is limited with pre-scripted puzzles; it is thus unknown how it would scale both to the complexity of planning used and whether the characters would keep functioning as expected from the goals that the story director hands them.

**Table 3.3.12** OPIATE metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 3 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 1 | 2 | 1 |

**3.3.13 Proactive Persistent Agent Architecture (PPA)** [MDC03, MC03, Mac04]

PPA is an architecture to simulate human-like behaviour for virtual agents. Their aim is that the agents can pursue their goals regardless of the actions of human users. They use architecture to control the agents comprised of three parts; the Schedule Unit, the Role-Passing Unit and the $\mu$-SIC System (or Social Unit). Using a shared knowledge base, the agents and the components have an awareness of the immediate surroundings and objects and persons.

In the words of B. MacNamee, the general idea is that the human user should be able to: "observe a single character as, for example, she begins her day at home, goes to work, visits a bar and, finally, returns home again" [MDC03, p. 2]. To accomplish this, they pass the agents' roles that inform the agents what they should be doing because expected actions change during the day, according to a scheduled. According to its schedule, an agent moves from one scene to another, e.g. leaves homes and goes to work. The agent has an appropriate role, e.g. the "at work role" when it is at work according to schedule.

**3.3.14 Scenejo** [SWM06, Spi07, SS09, SH10, Spi09]

Scenejo uses a chatbot technology A.L.I.C.E. [ALI06] to create conversational games.

**Table 3.3.13** PPA metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 1 | 3 | 1 | 3 | 4 | 4 | 3 | 4 | 4 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 |

A Youtube example of an interactive drama, the office brawl, generated by Scenejo can be seen by following this link `http://youtu.be/cIf86hmsz4E`. Scenejo generates the actions and responses of characters that a player can talk to using a free-text interface. The actors also initiate dialogues and events.

In the office brawl, there are all the essential elements that one expects in a drama. There are actors, a spectator; there is enactment, an imitation of human behaviour, mimesis [Ess76]. There is an inciting event, i.e. being told that there is an argument and the player needs to help calm their colleagues down. There is a complication as the argument heats up, and the characters start to be more aggressive. There is some climax when both of them start to recognise that they need to be more civil, listen to each other and then finally a breakdown when Ben gives up and refuses to continue working in the group. This form of drama is called a tragedy. The tension is due to the plot graph that Scenejo has.

The authors can directly author the plotline or leave it to Scenejo to generate it. They can use either AIML: Artificial Intelligence Markup Language [ALI06] as the bots dialogue base or author it themselves. Scenejo uses scenes to structure the drama; the scenes are sequential; they occur one after another. The scenes compete with each other. Sometimes many scenes are applicable, and then one needs to be chosen over the others, producing varied outcomes and increases replayability.

Stimulus-Response Element (SRE) in Scenejo allows the author to plan dialogue lines by an intermediate abstract structure. An editor allows the authors to design the SREs.

Each actor gets their dialogue graphs for each schene and the dialogue in AIML, this combined makes up the knowledge base of the actors.

This architecture requires considerable authoring; authoring every SRE in the SRE editor that is unintuitive for authors. It is also intractable for long dramas and is why the authors aim at making it more abstract and reusable [Spi07, SS09, SH10, Spi09].

**Table 3.3.14** Scenejo metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 3 | 1 | 2 | 1 | 1 | 3 | 2 | 1 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 1 | 2 | 2 |

**3.3.15 TEATRIX** [MPG00, MPP00, PMP00, PMP01b, MPB01, MPP01, PMP01a, PPMG02, PPM+04]

In the words of I. Machado & Co:

> *" children create the stories using a set of pre-defined scenarios and a set of pre-defined characters. These characters may act on behalf of the children or autonomously. "*
>
> – [MPP00, p. 1]

An agent has five components according to I. Machado & Co [MPP00, p. 5]:

- Perception Filter - determines the relevance of an agents perception and stores relevant information.
- World Model Update - stores any change in the world module.
- Emotional Reaction - to any emotional influencing event.
- Goal Update - events and emotions influence goals.
- Action Planning - plans actions that fulfil current goals taking into account all relevant detail.

When the child controls the agent, then the mind is mostly inactive, although if the child makes the agent do something that goes against the character that the agent is playing, the character will be upset. When the child is not playing the character, then the agent takes over.

There is a director that directs the agents; he even takes them over and controls them sometimes, although only when a child is not playing the agent.

According to R. Prada: "TEATRIX have six different roles for the characters: villain, hero, helper, magician, beloved one and beloved relatives" [PMP00, p. 3]. Based on roles from Propp's Morphology of the Russian Folk-Tale [Pro68].

In the words of I. Machado & Co: "Each one of these roles has a set of Propp's functions associated with it, and those functions create goals, behaviours and actions. For example, a villain must perform villainy."[MPP01, p. 3].

According to A. Paiva & Co, the details of the roles are [PMP01a, p. 4]:

- **Villain** villainy to disrupt peace and cause harm, specifically against the hero.
- **Hero/Heroine** seeking love, fortune or revenge on the villain.
- **Helper** is there to help the hero.

- **Beloved one and Family** frequently the victim of the villain or someone the hero is trying to please.
- **Donor or the provider** gives the hero the magic sword or secret info that allows the hero victory over the villain.

TEATRIX became E-TEATRIX [ZPTP03, ZPPZ04]; they describe autonomous and goal-driven agents perceptive to the environment with scarce technical details. The actor agents apply Propp's functions defined in Propp's Morphology for the Russian folktales [Pro68]. The agents' act in a sequential rather than autonomous manner. They decide their next action based on Propp's scripts. They use a planning algorithm for the scripts, in the words of A. Paiva:

> The planner uses a partial order planning algorithm together with a set of pre-defined plans associated with each role and function of the personae.[PMP01b, p. 6]

**Table 3.3.15** TEATRIX metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.16 The Virtual Storyteller [SV07, SVB07]

The Virtual Storyteller creates emergent stories from character interactions, emphasising emergent improvisational theatre that a narrative agent tells.

There is a plot agent to ensure proper plot development. Autonomous character agents that have individual emotions and beliefs inhabit the story world, and by their interactions, a story emerges. The characters use techniques from the improvisational theatre, and a world agent keeps track of the current world state. They use natural language processing of the story and create a synthesis. They developed rules to transform the synthesised speech into storyteller speech, e.g. with the expected emphasis that a storyteller will use to provoke suspense and excitement.

**Table 3.3.16** The Virtual Storyteller metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 2 | 3 | 3 | 2 | 1 | 1 | 1 | 4 | 4 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.17 The Virtual Theatre Project [HRSB+94, RHR96, RHR97, RHR98]

The Virtual Theatre Project use what they call directed improvisation, meaning that improvisational actors receive directions or constraints that they use to centre their improvisation on. For example, an actor can be told to walk to a table but not how to do it. The actor can rush to the table when playing an energetic character and shuffle when portraying laziness. They tried it in Computer Animated Improvisational Theatre (CAIT) which aimed at children between 9 - 12 [HRSB+94]. In Cybercafé [RHR96, RHR97, RHR98] Which has two or more players as customers in a café interacting with an autonomous waiter. In a Master-Servant scenario, servant Gregor through a series of postures, switches places with his master Otto and becomes master himself [HRvGH97].

**Table 3.3.17** The Virtual Theatre Project metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 2 | 1 | 3 | 1 | 4 | 1 | 1 | 1 | 1 | 1 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.18 Erasmatron [Cra03, Cra04]

Erasmatron is a story scripting system to script actions as 'verbs' and connect them in a sequence to account for any eventuality. There is an apparent unsolved intractability in scripting for every eventuality, and the work has yet to prove successful — current text-based stories generated last less than a minute.

**Table 3.3.18** Erasmatron metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 3 | 1 | 2 | 4 | 3 | 4 | 1 | 2 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.19 MARGIE [SGRR73, SGIR75, SA77, SLB80, SK87, SBB+90, SKe+94]

The Memory, Analysis, Response Generation, and Inference in English is a set of programmes and algorithms, in the words of R. C. Schank:

> *" analyse natural language sentences into an underlying language-free conceptual representation; stores this representation in memory, establishes references and makes inferences encodes the con-*

*ceptual representation into an English syntactic representation; and then encodes the syntactic representation into an English string*
”

– [SGRR73, p. 255]

Inference means that the program can draw logical inferences from natural language (NL), see Example 3.3.1:

**Example 3.3.1** *Inference mode: [SGRR73, p. 255], [SGIR75, p. 310]*

- *INPUT: John told Mary that Bill wants a book.*
- *OUTPUT1: A book about what?*
- *OUTPUT2: Mary knows that bill wants a book.*
- *OUTPUT3: Bill wants to come to have a book.*
- *OUTPUT4: Bill wants someone to cease to have a book.*
- *OUTPUT5: Bill wants to read a book.*

MARGIE can also paraphrase NL as seen in Example 3.3.2:

**Example 3.3.2** *Paraphrase mode: [SGRR73, p. 255], [SGIR75, p. 311]*

- *INPUT: John killed Mary by choking Mary*
- *OUTPUT1: John Strangled Mary.*
- *OUTPUT2: John Choked Mary and she died because she not could breathe.*
- *OUTPUT3: Mary dies because she was unable to inhale some air and she was unable to inhale some air because John grabbed her neck.*

Schank’s programme is a Natural Language processor; for example, the sentence “John told Mary that Bill wants a book” when the programme considers “told” it infers the following:

*“ TOLD: The analyzer uses verbs to make most of the important predictions upon which analysis proceeds. ’Told’ maps into MTRANS conceptually. MTRANS makes conceptual predictions about what kind of objects are likely to follow it. For example, MTRANS predicts that there will be a human actor, some information as object, and a recipient that is a part of the mental apparatus of some other human. The tense of ’told’ tells us the MTRANSing took place before the time of speaking. ’Told’ also indicates that the subject of the verb is the actor of the action and*

> *that the information that was 'told' was originally in the active part of the actor's mind, which we have labelled the Conscious Processor (CP). We know also that if a human follows 'told' then that person's CP is the conceptual recipient of the MTRANS. 'Told' predicts that the information needed as the object of the MTRANS will follow the human who is the syntactic object. "*
>
> – [SGIR75, p. 315]

The 'MTRANS' seen in the above quote is one of MARGIE's primitive acts. The MARGIE analyser parses each sentence into a contextual dependency representation rather than a syntactic one. As a result, MARGIE can have a conceptual understanding of NL. The analyser has a knowledge base that is built conceptually from the NL that they receive as input.

The problem with MARGIE and similar systems is that parsing sentences in this manner is time-consuming, even in small systems, and as the input and system grow, it becomes intractable, so Schank introduced scripts.

Schank's scripts are abstracting conceptual information from blocks of text rather than parsing it word for word. For example, the following passage from the New York Times:

> *" An Arabic speaking gunman shot his way into the Iraqi Embassy here (Paris) yesterday morning, held hostages through most of the day before surrendering to French policemen and then was shot by Iraqi security officials as he was led away by the French officers. "*
>
> – [SLB80, p. 80]

In this passage, the word 'gunman' immediately initiates three possible scripts; $ROBBERY, $TERRORISM and $KIDNAP. The program starts to fill in the scripts to see which matches the text, and the word 'hostages' narrows it down to $TERRORISM. Now the program can match actors, goals and actions from the text to the script. The scripts speed up the natural language process and additionally enables object-oriented mapping of inference rules and predicates in Schank's programs. There is no narrative structure inherent in MARGIE; hence those metrics are all at 4 in Table 3.3.19.

**Table 3.3.19** MARGIE metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 3 | 2 | 2 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

#### 3.3.20 Tale-Spin [Mee81]

Tale-spin applies first-order logic to generate tales. It uses Schank's scripts and primitives to create event plans in the narratives generated. Tale-spin creates stories from what they call rationally behaving characters. The characters are rational in that they follow first-order logic to reach goals. They use a problem solver that creates subgoals from main goals to solve the main goal by first satisfying the subgoals. The problem with this type of methodology is the intractability of creating the vast logical knowledge bases needed to generate sizeable output. They created several small tales, for example "The fox and the crow". The fox, using flattery, tricks a crow into letting go of a cheese it is holding in its beak by getting the crow to sing. Even in such a small tale, they had considerable problems with the errors created if the knowledge base created unwanted knowledge. For example, if the crow ate the cheese before the fox came, it realised it was holding a cheese in its beak.

**Table 3.3.20** Tale-Spin metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 3 | 2 | 2 | 4 | 4 | 4 | 4 | 4 | 1 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 |

### 3.3.21 SAM [Cul77b, Cul77a]

An experiment in using Schank's scripts to reads a newspaper and then be asked about what it said, Example 3.3.3:

**Example 3.3.3** *Question-Answering :*

- *Ql : Who went to China?*
- *Al : NADIA AND ENVER HOXHA WENT TO COMMUNIST CHINA.*
- *Q2: How did Enver Hoxha and Mrs Hoxha get to China ?*
- *A2 : THEY FLEW TO IT .*
- *Q3: Why did they go to China?*
- *A3 : ENVER HOXHA AND HUA KUO-FENG WANTED TO DISCUSS CHINA ALBANIA ECONOMIC AFFAIRS.*

**Table 3.3.21** SAM metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 3 | 3 | 2 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

### 3.3.22 Belief-Desire-Intention (BDI) [Bra87]

BDI uses psychological views on how humans perform practical reasoning. The BDI describes a rational agency [CL90, RG91] which assumes that humans strive to fulfil specific rational desires as long as they are attainable. If the desired element becomes inactive or unattainable, then the agent stops perusing that goal. Work on formalising BDI in the '90s has given rise to several BDI-Based programming languages such as 2APL [Das08], Jack, Jadex and Jason [BDDF05]. Examples of implementations in agents include; encoding a goal-risk framework into the Jadex platform so that an agent's deliberation process considers risks and associated costs [AGZ07] and BDI based virtual agents with a theory of mind [SDM09].

**Table 3.3.22** BDI metrics

| Drama | GT | OI | SI | FD | PS | ST | DT | PR | PI | AN | NI |
|---|---|---|---|---|---|---|---|---|---|---|---|
| = | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 1 | 4 |
| Agent | LR | SI | Ra | TM | Em | CT | DA | Du | KB | TE | Le |
| = | 1 | 2 | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 1 |

## 3.4 Summary

We show the average score for drama and agent metrics in Table 3.4.1.

**Table 3.4.1** Player's experience evaluation

| System | Drama avg. | Agent avg. |
|---|---|---|
| 3.3.1 BARDS | 2.27 | 1.82 |
| 3.3.2 Mimesis | 2.64 | 2.0 |
| 3.3.3 IN-TALE | 2.09 | 2.0 |
| 3.3.4 IDA | 2.55 | 2.0 |
| 3.3.5 Thespian | 2.27 | 1.55 |
| 3.3.6 IDtension | 2.27 | 2.0 |
| 3.3.7 Façade | 2.45 | 2.0 |
| 3.3.8 U-DIRECTOR | 2.73 | 2.0 |
| 3.3.9 FAtiMA | 1.45 | 1.73 |
| 3.3.10 GADIN | 1.27 | 1.82 |
| 3.3.11 NOLIST | 2.09 | 2.0 |
| 3.3.12 OPIATE | 1.55 | 1.73 |
| 3.3.13 PPA | 2.64 | 1.91 |
| 3.3.14 Scenejo | 1.64 | 1.82 |
| 3.3.15 TEATRIX | 1.27 | 2.0 |
| 3.3.16 V-Storyteller | 2.09 | 1.91 |
| 3.3.17 V-Theatre | 1.73 | 2.0 |
| 3.3.18 Erasmatron | 2.82 | 2.0 |
| 3.3.19 MARGIE | 3.45 | 2.0 |
| 3.3.20 talespin | 3.18 | 1.82 |
| 3.3.21 SAM | 3.55 | 2.0 |
| 3.3.22 BDI | 2.64 | 1.36 |

The goals of related systems are many and diverse; this Section briefly discusses our goals and to what degree they are reflected by other systems in the field.

Compellingness; H. Barber says:

> *" is defined to include all required features of an interactive narrative. The compellingness criteria must be satisfied by an interactive narrative system. "*
>
> – [Bar08, p. 19]

Replayability: if a game that can be played often yields greater value for the player who can enjoy it for longer. A common goal is that the players should experience a novel plot each time. Therefore. emergent systems rarely have the exact same plot for two individual runs and offer greater replay-ability and variety.

This is where the weakness lies in emergent systems; not only is the ending unknown, but it can also be indeterminable. It can be difficult to keep the emergent story on track to create a structured story. The characters often start doing something unexpected or inappropriate that does not conform to the expectations of the players. This is, for instance, a worry of the Virtual Storyteller. Emergent systems, therefore, use narrative functions to control how the drama emerges.

Divide and conquer: Many systems apply some form of dividing the story into many small parts or tasks and use planning algorithms, for example, STRIPS [Byl94] to plot the story ahead of the player [Bar08].

Emergent systems rely on interaction for the drama to unfold, see Section **??**. FearNot!, Gadin, Nolist, OPIATE, PPA, Scenejo, TEATRIX, The virtual storyteller, and The virtual theatre project are examples of emergent systems. Not all emergent systems have players; for instance, the Virtual Story Teller relies on the interaction and emotional models of its NPCs.

Protagonist, Embodiment: A prominent design goal is that the player becomes so immersed in the drama that she emphasises with the character that she is playing and consciously or subconsciously becomes embodied in the character. For instance, a player playing a ranger in an RPG senses herself as being that ranger and thus has the ranger behaving as she would behave if she was the ranger. It is even better if the player becomes the protagonist of the story identifying with the role such that she really feels that she accomplished to resolve the complication of the drama and that the drama was centred around her character. Preferably this happens when the player is pursuing her own goals.

Immersion & Believability: Most games aim for high levels of immersion, so that the player spends a large amount of time playing it, and becomes emotionally affected.

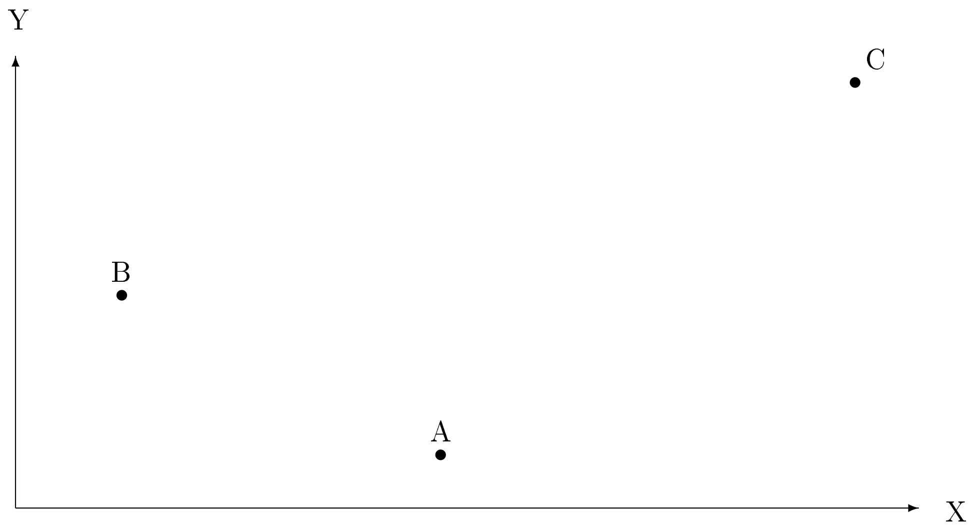
According to E. Brown and P. Cairns, there are three levels of Immersion [CCB+06]:

1. Engagement, the player is willing to play the game, learn the controls and such, and is willing to invest time in the game to advance towards the end or goal of the game.
2. Engrossment, when the player is not only engaged but is also emotionally affected by the game.
3. Total immersion, when the player becomes one with the game, completely losing awareness of their immediate surroundings.

A significant part of immersion is believability in the characters and the game world, and an empathy with the dilemmas they face. In the words of A Paiva & Co:

> *" In particular, the expression of emotions makes these characters more human-like and believable. ... there are three important points when expressing emotions:*
>
> 1. *the emotional state of the character must be clearly defined, in such a way that is undoubtedly perceived by the viewer;*
> 2. *the emotional state affects the reasoning process, and consequences must be perceivably reflected in the actions of the characters; and*
> 3. *emotions can be accentuated or exaggerated, to clearly communicate the viewer the emotional state of the character. Another element is personality. A coherent character, that acts according to its personality*
>
> *will be more believable than a character that has no long term coherence in its behaviour. Thus it is not so much one property or another that matters but the combination of all these factors, that together providing ingredients for the building of the Believability believability in a synthetic character. "*
>
> – [PDS+04, p. 2], [PDS+05, p. 237]

**Figure 3.4.1** believability, the X axis is believability, and the Y axis is complexity




We can plot Believability believability on a two-dimensional scale as seen in Figure 3.4.1 where the Y-axis is the level of complexity, and the X-axis is the level of believability.

 95

In the Figure, the (C) is where both believability and complexity is so high that it is indistinguishable from a human player.

Current computer games have virtual agents that aim for high believability and low complexity, see (A), in order to optimise available resources.

We aim to increase complexity at the cost of believability, see (B). Future research could optimise on this research to incorporate greater believability while retaining complexity. The increase in complexity is to create actors that play complex characters with advanced levels of emotional reactions, influenced by traits that affect the characters core responses.

**Drama awareness** We are, therefore introducing rational actor agents that manage the drama similar to improvisational theatre. For this, the agents need to be aware of the drama progress and have a complex reasoning process to be able to collaborate and respond rationally.

**Decentralised management** In order for this to be applicable to modern, fast-paced computer games that run on the large distributed systems, we introduce our decentralised drama management.

# 4

# Directed Emergent Drama (DED)

## 4.1 Introduction

The Directed Emergent Drama (DED) is a decentralised drama management engine for interactive emergent dramas in computer games. The engine's architecture includes a director agent, actor agents and schemas that guide the emergence of an interactive drama. The drama emerges from the interactions of actors and players. We have 6 publications of this work:

1. Schemas in directed emergent drama, in *proceedings of the 1$^{st}$ Joint International Conference on Interactive Digital Storytelling ICIDS08*, Erfurt, Germany, 2008. [AK08]
2. Directed emergent drama vs. Pen & Paper Role-Playing Games, *AISB'09 Symposium: AI & Games*, Edinburgh, UK, 2009 [AK09a].
3. Duality of actor and character goals in virtual drama, *9$^{th}$ International Conference on Intelligent Virtual Agents*, Amsterdam, Holland, 2009 [AK09b].
4. A critical review of interactive drama systems, in *AISB'09 Symposium: AI & Games* , Edinburgh, UK, 2009 [ABK09].
5. BNs: real-time applicable decision mechanisms for intelligent agents in interactive drama, *proceedings of the IEEE Conference on Computational Intelligence and Games (CIG2010)* , Copenhagen, Denmark, 2010 [AK10].
6. Believable Bots, in book *Actor Bots*, P. Hingston, 2012 [AK12].

In this chapter, we describe the DED in detail and its implementation, followed by a discussion of how it compares to related work.

In Chapter 2 we introduced constructs and technology that our work builds on in Sections:

2.2 Narratology; Structuralism and Drama theory

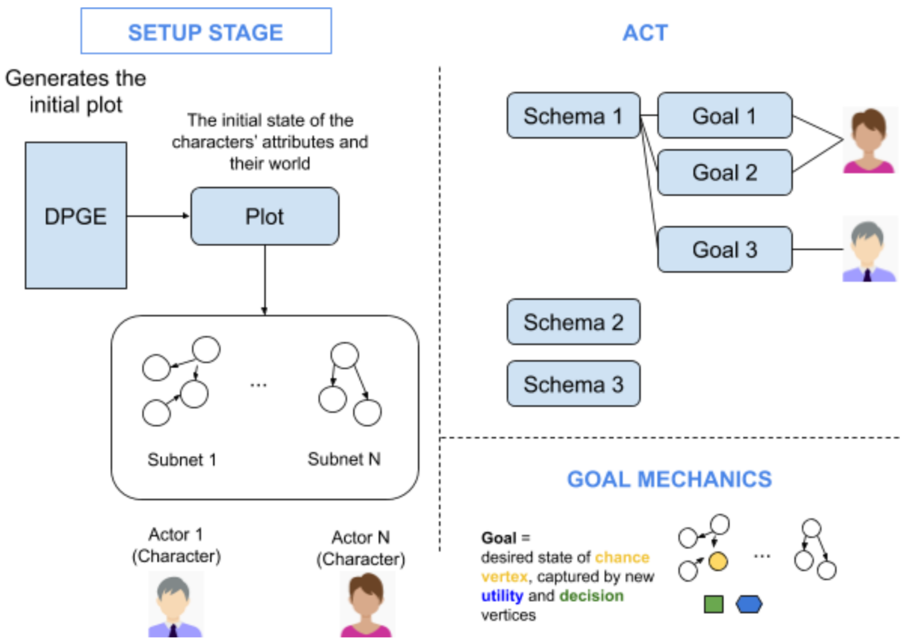
2.3 Computation; BNs and GT and how the two are combined in the MAIDs to calculate a Bayesian equilibrium. that the NPCs use for rational reasoning.

2.4 Former work; published in the articles listed above.

We collaborated with Improbable's Enterprise department and connected the DED to their proprietary multiserver platform Spatial OS. Improbable describes Spatial OS, on their web, as:

> " *a cloud platform that provides hosting, online services, tools and a multiserver networking stack for developing and operating multiplayer games.* "
>
> – [Imp21a]

**Figure 4.1.1** $a_1$ is evaluating how $c_1$ expects that $c_2$ will react to $5_0$, displayed in the GeNIe UI [DSL07a].




The DED is relevant to the Improbable Defence current product, the Single Synthetic Environment for Defence. In this Chapter we use two scenarios to explain the technology:

1. The murder mystery introduced in Section 2.2.1.4

2. An academic scenario of an aid mission in Afghanistan called the Afghan poppy scenario and is about an officer deciding whether to give aid to an Afghan farmer depending on whether the farmer is, or intends to, grow poppies, see Section 4.2.

Both of these scenarios are implemented and the code for them is available in our GITHUBLINK.

We now give a high-level view of how the DED drama process is initiated, and we refer to Figure 4.1.1 for visual guidance.

1. We start by generating a plot with the DPGE, introduced in Section 2.4.1. The DPGE instantiates every vertex to a randomly selected state in an Object-Oriented Bayesian Networks (OOBN), introduced in Section 2.3.1.6. These instantiated states represent the plot that defines the characters' attributes and their world's initial state.
2. Knowledge base: The OOBN contains several subnets that define the specific personal knowledge of each character. Each actor also has a copy of the entire uninstantiated OOBN as a blueprint of the environment's causal structure, which corresponds to Harsanyi's exogenous data of the game, introduced in Section 2.3.2.3. The actor uses the blueprint when reasoning about their world. The actors also receive the knowledge specific to them as a list of vertices and their states. For example, a farmer receives the knowledge specific to his person and his farming, including the state that he is growing poppies on one or more farms. The farmer's knowledge is, for the most part, only known to him unless it has a causal connection to other character subnets. For example, if the farmer is growing poppies, it is likely that the army has destroyed his crops and knows about his poppy farming. The causal structure of the OOBN facilitates this inference. For example, if the DPGE sets the chance vertex for the destroyed crop to true during plot creation, then that evidence is propagated to the farmer's chance vertex for the destroyed crop by updating the OOBN. Therefore, the characters start with a knowledge base that is specific to them and any knowledge that they can infer by instantiating their OOBN blueprint and updating it. Once the plot and actors are ready, the drama can start.
3. The drama is in several acts; each act has several schemas, see Figure 4.1.1. A schema has a collection of goals attached to one or more character roles, for example, a farmer or an officer. There are two types of goals;
   (a) Drama goals that ensure the dramatic arc requirements.
   (b) Character goals that ensure that the character remains IC.

   The director uses the drama goals to maintain the dramatic arc and will only move the drama to a new act when all drama goals for the previous act are satisfied. The purpose of this goal duality is to cause tension and conflict in the drama. The actors know that they are playing characters in a drama and are responsible for satisfying drama goals. They do this by requesting aid from other actors, without the player knowledge, and pass the drama goal over to the other actor.

A goal in itself is a simple object that can have massive emergent consequences by leveraging the structure and game-theoretic reasoning of the MAIDs. A goal references a subnet, vertex and a state or no state as appropriate. A goal has the following attributes:

- The goal always references a subnet and a vertex.
- The goal may have a specific state, for example, when giving information such as “destroyed crop" = “true". Alternatively, no state, for example when asking a question: “are you growing poppies?" has the state none since it is not known.
- Identity made of $subnet\_vertex\_(state|none)$.
- It has a type for the actor to know how to reason about it.
- It has a set of functional roles, e.g. a goal for an officer or Officer Commanding (OC).
- It has an aim role, i.e. ask a farmer.
- The director can assign it as part of a schema.
- The actor can create it as part of its reasoning process, for example, if they have the goal of gaining information before making a decision based on that information. They will generate their own question goals based on vertices connected to the decision chance vertex.
- It can be passed to the actor from another actor when requesting aid due to drama and character goal conflict.
- The actor creates decision and utility vertices on-the-fly when creating a MAID to evaluate an optimal action.

Once the goals are distributed and the BNs of each character updated accordingly, the drama can start. We implemented the murder mystery both as a turn-based game and as a real-time asynchronous game. However, the poppy scenario only has an asynchronous implementation.

This chapter is organised in the following sections:

4.2 Poppy Scenario: The Afghan poppy scenario.

4.3 Schemas: The schemas that are used to structure the emergence of the drama with a rise and fall in conflict and suspense.

4.4 Director: The director which uses the schemas to guide rather than direct the drama.

100

4.5 Actors: The actors, their architecture, knowledge base, and rational reasoning.

4.6 Players: The player's role and interaction with the game and benefit from the actors AI.

4.7 Creating a new drama: The drama design and authoring process.

?? Comparison to background and related work: The work in this chapter compared to the work introduced in Chapters 2 and 3.

4.8 Summary.

## 4.2 Poppy Scenario

We have implemented an academic scenario using Improbable's SpatialOS. We collaborated with Improbable Defence, currently developing a Single Synthetic Environment (SSE) for Defence. We are looking at the DED and the SSE training scenario's compatibility and have implemented an Afghan poppy scenario, an entirely academically sourced motif, to do initial testing. The scenario does not represent current product work at Improbable.

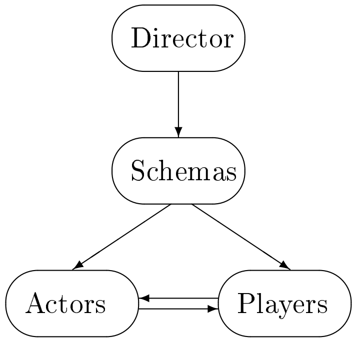
Head of Enterprise research, Christoforos Anagnostopoulos has supervised this effort.

The drama is divided into as many acts as is required to pace the emergence. For example, in the Afghan Poppy scenario where the officer is the player, we designed the following five acts:

1. The officer receives orders and information from the OC regarding an Afghan farmer. The information is any relevant intelligence that the OC knows. The orders are to investigate whether the farmer is, or intends to, growe poppies.
2. The officer meets the farmer and starts to ask the farmer about his farms and whether he grows Poppy. The farmer wants aid from the UK forces and has a goal to conceal his farming of Poppy; believing that it lessens his chance of receiving aid.
3. Officer feeds back to the OC what he learnt from the farmer and from anyone else he spoke to that revealed relevant information. The OC again gives the officer any relevant intelligence and orders him to speak to the farmer and offer aid if possible, i.e. if the farmer either is not growing poppies or intends to stop growing poppies.
4. The officer meets with the farmer, and using his judgment, based on his interactions with the farmer, will either offer aid or decide against it.
5. In the final act, the officer relates his decision to the OC, which gives his view of whether he thought the officer made the right call or not.

## 4.3 Schemas

**Figure 4.3.1** The DED architecture




The DED engine has a director agent, actor agents and schemas that guide the emergence of an interactive drama. The drama emerges from the interactions of actors and a player, see Figure 4.3.1.

Schemas contain all information from the director to the actors. The director uses the schemas to direct the emergent drama, and the actors use schemas to orient themselves to where the drama should emerge.

The director agent sends schemas to the actor agents and players. The schemas are motif like structures that a director uses to inform the actors of where to aim the drama emergence. Motifs are generic structures that do not have a narrated list of events. They are situations in the words of N. Szilas:

> *" In theatre and drama, the concept of situation is a key dimension in the analysis of a given piece. A situation is understood here as a set of characters and their relationships that makes the drama interesting. Examples of dramatic situations include the 'love triangle' —two characters love a third one—, forbidden love between siblings (who may ignore their family ties), love between different social classes, and others just to mention the romantic domain as example "*
>
> – [SER16]

In conflict areas like Afghanistan, it is easy to see how the above list is relevant, repeated wrongs by far superior force, lost their kin to the cause and even seen them brutally slain before them by ambitious rivals that covet an elusive prize. People make mistaken judgments or lose their honour in the heat of the moment.

The schemas function is to assign goals to the actors that create this type of conflict. For example: In Section 2.2.1.2 we introduced a list of G. Polti's [Pol21] situations which the numbers below refer to, an Afghan farmer has

 103

been growing poppies against his religion (27, 33, 34), and he does not want the UK forces to know. He may have lost a loved one or hopes to get the UK forces to help find them (35,36). Perhaps his brother or son have gone to war, and maybe he had a part in encouraging them (23, 33, 34). There may be a smuggler or an Al Qaeda agent present competing with the UK forces for the farmer's alliance (24, 30). The schemas' goals can be specific down to particular knowledge gain, for example, finding out if the farmer is growing poppies. They are also more general; for example, the farmer seeks aid from the UK forces. The schemas do not contain any instructions on how to accomplish the goals.

A schema has three components:

1. *Acts*, a list of applicable acts for the schema.
2. *Roles*, a list of applicable roles for deploying the schema.
3. *Goals*, a list of drama and character goals for the actors and players.

### 4.3.1 Acts

Pace the drama: Schemas that have goals relevant to a specific act have the number of that act in their list of acts so that the director can quickly find relevant schemas.

For example, schemas for the five acts in the Poppie scenario are:

1. Act: Orders and information.
2. Act: Question farmer.
3. Act: Feedback and orders.
4. Act: Aid.
5. Act: Review.

The actors only have goals active for the current act's schemas. For example, the OC and officer initially receive the OC orders and OC information. When the OC gives the officer orders, he gives the orders as a schema assigned to act II.

The officer receives the schema and the goal to question the farmer. However, it is not a current act schema. Therefore, the schema's goals do not become active for the officer, and he doesn't act on them until the director signals act II.

Otherwise, the officer could run off and question the farmer without receiving all of his orders and information from the OC.

The resulting structured pace of the drama is evident in the 4 years of available logs that we have uploaded to the departments GitHub repository.[1]

[1]https://github.com/uoy-research/DED/tree/main/SecondLife/logs

**Dramatic arc** In Section 2.2.1.1, we introduced Freytag's pyramid [Fre63]. In Section 2.2.1.3, we introduced Propp's theory of a uniform structure and predictable patterns [Pro68]. In Section 2.2.1.4, we combined Freytag's and Propp's structuralism with the work of [Tod77, Pri03, Kno92] on the structure of the murder mystery and the narrative.

We implemented the three-act murder mystery structure in Section 2.2.1.4, and the intended pyramid structure is evident in the logs.

Similarly, we designed schemas for the Poppie scenario to start on a nonconflicting and straightforward act of giving information and orders. The introduction act is followed by the officer tries to understand whether the farmer is growing poppies and whether they can offer him aid and gain an ally. The tension builds because the officer doesn't know if the farmer is telling the truth. In some cases the farmer is farming Poppy and sometimes not. The officer may need to do some investigative work to learn the truth.

The climax is when the player must decide whether to aid the farmer or not, with out having complete information and must make an educated guess. The resolution is a debrief with the OC. This structure can be seen from the logs in the Departments GitHub repository.[2] This is a scene that we found thrilling due to the multiple parties that we could introduce and the complex political and cultural situation involved.

### 4.3.2 Roles

The drama has primary roles assigned to the plot and actors are allocated these at the start. In the Afghan Poppy scenario, these roles include:

- Officer, the protagonist, played by the player.
- The OC that incentivises the officer, similar to a quest giver in computer games.
- A farmer that is growing poppies but wants aid from the UK forces.
- A travelling merchant that is selling things near the farm.
- An Al-Qaeda relative of the farmer that is affecting the farmer's decisions.

These roles can have several secondary functions, including parent, partner, sibling, a believer in a cause or religion, victim, hero, rival, or lover. The secondary roles introduce conflict and are determined by the DPGE when creating the plot for the drama.

For example, the farmer can be a husband, a father and a brother and hope to find a lost relative, as in Polti's situation number 35.

Additionally, events can lead to characters adopting new roles; for example, the farmer may assume a judge's role and possibly passes a hasty judgment, as in Polti's situation number. 33.

[2]https://github.com/uoy-research/DED/commit/ee8f41b719e8b40744d9fae4b880505ca77e2e15

The roles are there for the director to understand which schemas the director can deploy and to which characters. For example, the officer has several giving and receiving orders goals, and the farmer has a receiving aid goal.

There are no specific actions or dialogue that comes with the character roles; the actors do not use the roles or the roles of others in their reasoning. For example, when judging if someone is trustworthy, they must use their current knowledge base and judge trustworthiness from experience.

On the other hand, we create the initial knowledge based on their roles, i.e. the farmer knows what a farmer would know, and they are assigned goals based on their roles.

Their goals also tell them which role to address to fulfil the goals. For example, the OC has the goal of informing the officer.

### 4.3.3 Goals

The goals are pivotal in the DED; everything that actors and players do is dependent on what goals they have.

Each goal has the following parameters:

Owner: Indicates whether the goal is for the character or the drama. Drama goals are goals that are necessary for the drama, while character goals guide the actor in how to stay in character. For example, the drama is about the officer, played by the player, discovering that the farmer is growing poppies and deciding whether to give him aid regardless. Conversely, the farmer's character goal is to hide his Poppy farming from the officer to maximise his chance of receiving aid.

The aim of having these conflicting goals is to introduce conflict and deception to the drama. The player must resolve the goals for the drama to progress. We require conflict and for the player, as the protagonist, to overcome challenges for the drama to reach a climax. There is conflict and disruption of status-quo in Polti's situations and Propp's morphology to build suspense.

Role: Indicates which role to assign the goal. There are generic goals that apply to multiple roles, and there are goals for specific roles.

Applies to: The applicable role for the goal. For example, the OC's has a goal of giving orders to 'officer' for him to give orders to his officer. Similarly, the officer has a goal of gaining information regarding Poppy farming from the role 'farmer'.

Aim: Indicates a vertex $v_i$ and a state $v_i$ that is in the actor's BN. For example, the farmer's goal to hide his Poppy farming is $v_{grows\_poppy}s_{false}$.

Importance: Goals are ranked by importance. It is, for example, more important to answer the telephone than to play the next card in a bridge game.

Satisfied: Indicates whether the goal has been satisfied.

Examples of goals:

- Murder mystery:
  1. Drama goal: Reveal motive.
     - Role: suspect
     - Applies to: detective
     - Aim: Motive = True
     - Importance: High
     - Satisfied: False
  2. Character goal: Hide motive.
     - Role: suspect
     - Applies to: detective
     - Aim: Motive = False
     - Importance: High
     - Satisfied: False
- Poppie farming
  1. Drama goal: Reveal Poppie farm.
     - Role: farmer
     - Applies to: officer
     - Aim: Farming Poppie = True
     - Importance: High
     - Satisfied: False
  2. Character goal: Hide Poppie farm.
     - Role: farmer
     - Applies to: officer
     - Aim: Farming Poppie = False
     - Importance: High
     - Satisfied: False

### 4.3.4 Summary

The schemas are motif like abstract structures similar to Polti's 36 situations [Pol21], that guide the actors with goals that the actor aims to fulfil in the course of the drama.The goals are matched to the actors by the role of the actor. They introduce conflict between character and drama goals and introduce tension and suspense.

The schemas have two types of goals; character goals for the actor to stay in character and drama goals to structure the drama.

There are no explicit instructions in the form of dialogues, gestures, actions or sequencing in the schemas they are assigned to specific acts for the director to pace the drama.

## 4.4 Director

The director is a shallow autonomous agent that is giving general directions in the form of schemas. The schemas guide the actors in developing the drama and engage the player in directed emergence by matching schemas to actor agents to align the drama to a specific drama genre. The director deploys the schemas introduced in Section 4.3, to guide the drama, depending on the drama's state, and keeps track of the state of the schema goals to read how the drama progresses and when to switch between acts and deploy new schemas. The director has no script and does not know what the actors are doing.

The director's view is, therefore, limited to an abstract understanding of the drama situation, and it cannot see the actions of the actors or the players, it sees:

- current act
- drama goal status
- character goal status
- actors are in the player's current location
- goals the actors and player have

Our target platform is MMOGs and huge simulations; Improbable developed SpatialOS specifically to facilitate vast game worlds by distributing the computation across multiple servers, see Figure 4.4.1, as they say in their documentation:

> *" it brings together many servers so they're working as one. But it does this in a way that makes a single world which looks seamless to players. "*
>
> – [Imp21b]

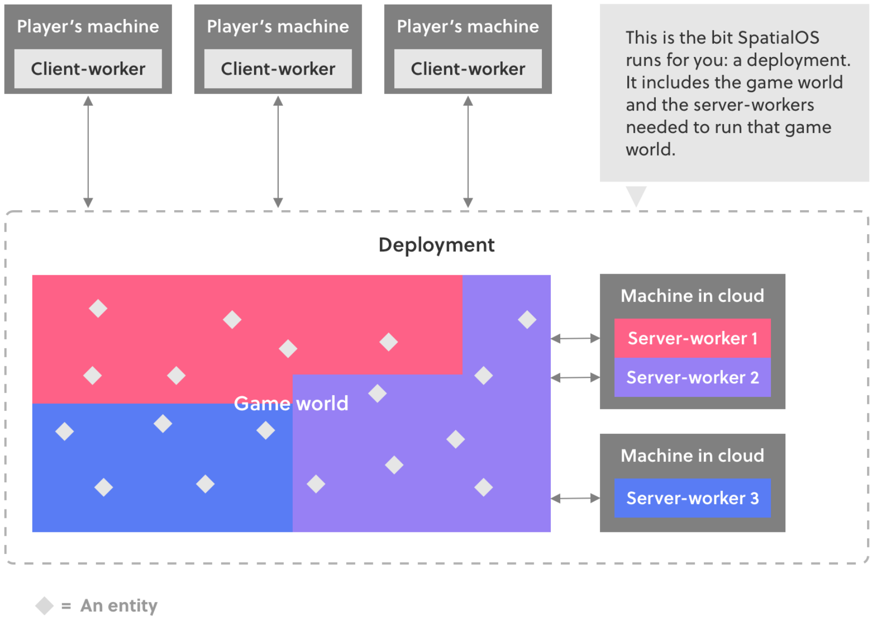
In MMOG, where the computation is distributed across multiple servers, having the director make all the decisions for each actor for all interactions that the actor has would it would rapidly escalate into an intractable computation problem, if for no other reason than the enormous amount of messages, as A. S. Tanenbaum et al. says:

> *" Even if we have virtually unlimited processing and storage capacity, communication with that server will eventually prohibit further growth. "*
>
> – [TS04, p. 10]

As discussed in Section 4.3, the schemas have two types of goals; character and drama goals, but no scripting, actions or direct instructions:

 109

**Figure 4.4.1** SpatialOS distributes computation across multiple servers to run MMOGs that are too large for a single server [Imp21b]




- Character goals for actors' to stay in character and to aim for role relevant objectives.
- Drama goals are for the director to evaluate the drama situation computationally.

The director deploys schemas with drama and character goals as required to keep up the level of complexity for each act.

The actors use their knowledge base and Game-Theoretic reasoning to control the drama, based on guidance from the director in the form of goals. The director:

- detects when the final act is complete and finishes the drama.
- moves the drama between acts when all the drama goals from the former act are satisfied.
- evaluates whether a schema is needed depending on the act and the state of the currently deployed goals.
- When a schema is needed the director picks a schema based on which drama goals are still left to fulfil and which schemas fit the current cast

Acts divide the drama up into small, manageable parts with few distinct drama goals. When all the drama goals of an act are satisfied, the director moves over to the next consecutive act. The character goals are less dependant on acts and schemas, linked to the characters and their roles. The character goals create conflict with the drama goals to create tension and suspense. Initially, the goals are simple but become more conflicting and challenging as the drama progresses.

In our Afghan Poppy scenario, the farmer wants aid from the UK and can have several other character goals. For example, he is resentful of foreigners invading his land and is making a profit from growing poppies.

The player's aim in a training scenario as an officer is to determine whether the UK should give him aid depending on whether he grows Poppy or not. The UK is willing to give aid to build an economy that is not dependent on opium production.

There is, therefore, a conflict between the farmer's goal of getting aid and the fact that he is growing poppies. The player's drama goal is to learn if the farmer is growing poppies and if he intends to do so in the future. There is, therefore, a conflict between these primary goals from the start. The farmer should reveal his Poppy growing for the plot progress while simultaneously keeping it hidden to receive aid.

Character goals are either deployed at the start as a crucial part of the main plot, for example, the farmer wanting aid. Additionally, they are in schemas as a counter to the schemas' drama goals to create conflict and variation. For example, a brother is part of the Taliban, and the farmer wants to stop growing poppies but is afraid to admit it in front of his threatening brother. Alternatively, he may have lost his brother in the war and not know whether he is alive or dead and be resentful toward the UK and foreigners believing that they killed him with their invasion. Perhaps he wants to find his brother and become an ally if he learnt that his brother was alive.

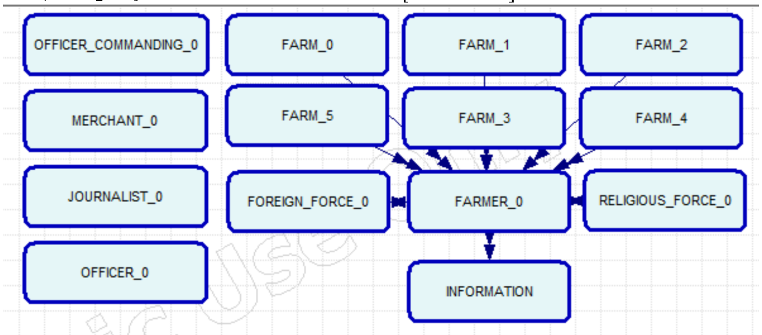
The director will deploy drama goal-related schemas at the start fo each act and then watch for them to be satisfied. He will also deploy schemas with new character goals if they fall below a minimum for that act. For example, at the start of the second act, the director deploys a schema with goals to ask the farmer about his Poppy growing. If the farmer has no character goal to complicate this, then the scene will become rather dull. Therefore, the director will select schemas with some character goals to increase the current drama score. When the director sees the number of character goals fall below a minimum for that act, it finds and deploys another schema.

### 4.4.1 Generating actor agents

The director also manages the start of the drama by calling the DPGE. The plot that the DPGE creates is a complete description of everything needed to create a knowledge base for each character.

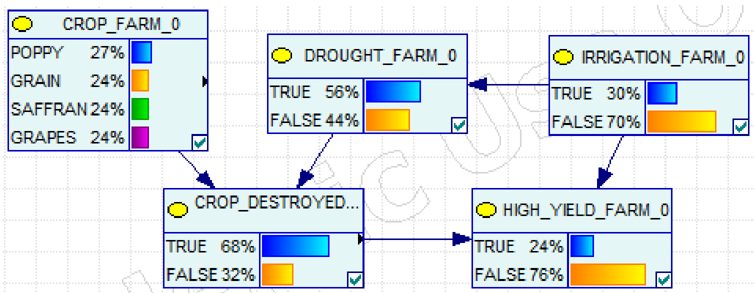
First, the director assigns each actor a drama role and then an initial knowledge base and a set of traits. It is not dependent on the plot, whether a character is gullible or honest. The traits are defined separately along with the initial character view of the scenario.

**Figure 4.4.2** An example of the Afghan poppy scenario knowledge base and its OOBNs, displayed in the GeNIe UI [DSL07a]




We use OOBNs, introduced in Section 2.3.1.6, to generate a knowledge base with minimal manual input. For example, in Figure 4.4.2, the OOBNs for an Afghan farmer with six farms or fields. Each farm is an object in the farmer's OOBN that is an instantiation of the same farm BN; see Figure 4.4.3 . The only difference between the farms is the instantiation of the vertices, which means that from authoring one farm BN the DED can randomly instantiate any variation of that BN; see Figure 4.4.4 the instantiated farm BN as having a destroyed Saffran crop due to drought. This means that we only design one farm and that one design is used for all similar farms, the values are randomly instantiated by the DPGE at the start.

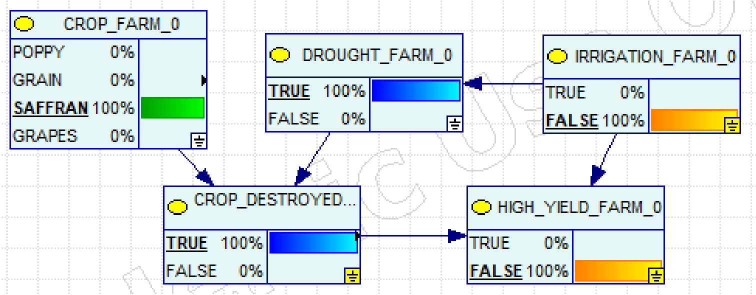
**Figure 4.4.3** A farm BN in an OOBN, displayed in the GeNIe UI [DSL07a]




The director assigns the parts of the plot that are relevant to each actor depending on their role. Each character in the plot is, therefore, created with the knowledge specified for their role. For example, the farmer knows everything in submodels ($FARMER_0, FARM_0, FARM_1, FARM_2, vFARM_3, FARM_4, FARM_5$), and the OC knows everything in the ($OFFICER\_COMMANDING_0, FOREIGN\_FORCE_0$) submodels.

When the actors receive their knowledge base, they instantiate the states

**Figure 4.4.4** An instantiated farm BN in an OOBN, displayed in the GeNIe UI [DSL07a]




in their OOBN according to their characters part in the plot received from the director.

## 4.5 Actors

The actors are the prima motor of the emergent interactive drama; they improvise a drama for the player based on guidance from the director.

**Decentralised drama management** The drama management is decentralised and distributed between the actors. The actors are responsible for the drama and will overrule the director's guidance if it is out of character (OOC) for their role. The actors work collaboratively, without the players' knowledge, to manage the drama when there are conflicts between the drama's requirements and the characters they are playing. The director has no part in resolving such disputes; the director is not aware of them.

**Drama awareness** The actors are aware that they are actors in a drama and need to ensure that the drama conforms to the expected genre. We introduce the duality concept between drama goals and character goals and how the actors collaborate to resolve this conflict while staying IC. For example, in the Afghan scenario, the farmer has a character goal of getting aid from the UK officer. Still, he has been growing poppies, which will decrease the chance of the UK officer offering support. The player's aim is to discover that the farmer is growing poppies and, therefore, the actor's drama goal is to reveal this.

When an actor makes and action that blocks a drama goal that the actor has, the actor will request aid from another actor to realise the drama goal without going OOC.

**Rational agents** The actors are rational in Game-Theoretic terms, see Definition 2.3.8, introduced in Section 2.3.2. We use causally structured BNs, introduced in Section 2.3.1.4 and MAIDs, introduced in Section 2.3.2.4 to reason about uncertainty, find a MAIDs equilibrium equivalent to Nash Equilibrium in an imperfectly informed game [KM03].

**Compatible with leading online game technology** We tested the DED with leading MMOG technology stack SpatialOS [Imp21a], coordinating agents in a single seamless game world across multiple servers. The rational, autonomous actors respond within a second to player actions, and the DED is fully integrable with leading MMOG technology.

**The structure of this section** We start by introducing the character traits that affect the actors' reasoning; see Section 4.5.1. Followed by introducing the structure of the actors' knowledge base and how it is populated, see Section 4.5.2.

We describe the actors' reasoning, starting with a simple task of creating and filtering appropriate actions to choose from, see Section 4.5.3. Followed by the act of informing another trusted character and using MAIDs, see Section 4.5.4, and responding to possible deception by challenge, threat or intimidation, see Section 4.5.4.2. We then describe how they can request from other actors and offer them aid, see Section 4.5.4.5, which facilitates actors' collaboration when their drama goals conflict with their character goals.

### 4.5.1 Traits

The actors need to show specific characteristics to be believable. The actors also need to choose from applicable sentences that are descriptive of the character they portray. There are many psychological theories and classifications of traits, and it is beyond the scope of this dissertation to discuss their relative merits and disadvantages. Our sole purpose here is to pick one such theory as an example and show how to map it to a BN to demonstrate how efficient the BNs are in handling such transient states' effects as mood and situation on the agents' decision making.

We experimented with two theories:

1. Cattell's 16 traits, see Section 4.5.1.1
2. Big five personality dimensions, see Section 4.5.1.2

#### 4.5.1.1 Cattell's traits

Cattell's 16 traits [Cat46] see Table 4.5.1.

Cattell selected these 16 traits after careful empirical research and he further spent a large part of his life developing examinations for these traits and the effect they have on behaviour (see for instance [Cat65]). Cattell also discusses how mood and situation can affect behaviour and his work plays a part in developing the big Five Factors of personality [Gol93] which are very broad definitions of personality that are thought to contain or represent the basic structure of all known personality traits. The big five factors are very well respected but as they are very broad we prefer Cattell's more narrowly defined traits as suitable examples. The big Five Factor personality model is of course the most well recognised and widely used personality model used today and DED would work equally well with it.

#### 4.5.1.2 Big five personality dimensions

The characters have traits based on the big five personality dimensions [RC03] that the actors use when reasoning. The traits are measured in terms of percentage value from low to high and represented as states in the BNs:

 115

**Table 4.5.1** Cattell's 16 traits

| Low range | High range |
|---|---|
| reserved | warm |
| concrete reasoning | abstract reasoning |
| reactive | emotionally stable |
| deferential | dominant |
| serious | lively |
| expedient | rule-conscious |
| shy | socially bold |
| utilitarian | sensitive |
| trusting | vigilant |
| practical | imaginative |
| forthright | private |
| self-assured | apprehensive |
| traditional | open-to-change |
| group-oriented | self-reliant |
| tolerates disorder | perfectionist |
| relaxed | tense |

- $COMPLETE$ 100%
- $HIGH >= 75\%$
- $MEDIUM < 75\%$ and $> 25\%$
- $LOW < 25\%$
- $NONE$ 0%

For example, the officer and OC have complete trust in each other.

We start by introducing the big five traits, and how some roles are likely measured high or low on the traits scale. When we say 'high', it means that this role would have a value of $HIGH$ to $COMPLETE$, and 'low' means state $LOW$ or $NONE$. If we do not specify a role as high or low for a trait, it means that it will fall in the midrange. Each trait affects specific parts of the actors' reasoning, which we discuss below and will go into greater detail later in the Section when describing individual reasoning elements.

#### 4.5.1.3 Neuroticism

> *" A general tendency to experience negative affects such as fear, sadness, embarrassment, anger, guilt and disgust. High scorers may be at risk of some kinds of psychiatric problems. A high Neuroticism score indicates that a person is prone to having irrational ideas, being less able to control impulses, and coping poorly with stress. "*

– [RC03, p. 69]

Religious people, terrorists, smuggler, and people negatively affected by the conflict will have high Neuroticism. Neuroticism affects the characters level of trust in others; they will have low trust and more likely to lie for no apparent reason because of excessive mistrust. Those high in Neuroticism are difficult to gain reliable information from, and they show a more significant response to force threats and intimidation.

In the reasoning algorithm, they have lower initial trust in others. They will have higher rewards for yielding to threats and intimidation, and it will affect their truth evaluation when stressed.

> *" A low Neuroticism score is indicative of emotional stability. These people are usually calm, even-tempered, relaxed and able to face stressful situations without becoming upset. "*
>
> – [RC03, p. 69]

Officers, journalist, and the merchant would be low in neuroticism.

#### 4.5.1.4 Extraversion

> *" Includes traits such as sociability, assertiveness, activity and talkativeness. Extraverts are energetic and optimistic. ... It was found that Extraversion is a valid predictor of performance in jobs characterised by social interaction, such as sales personnel and managers ... found a positive relationship between Extraversion and job performance of police personnel, and explained this relationship in terms of the high level of interaction in the police service. ... Extraversion is characterised by positive feelings and experiences and is therefore seen as a positive affect. "*
>
> – [RC03, p. 69]

Officers, journalist, and merchant are high in extroversion, it is a strong quality to have in a profession that requires a high level of interaction with people. High levels of extraversion will cause the character to say more, and be more trusting initially. They will ask about the person more than others.

> *" Introverts are reserved rather than unfriendly, independent rather than followers, even-paced rather than sluggish. "*
>
> – [RC03, p. 69]

Farmer, smuggler and terrorist are likely characters for introversion.

#### 4.5.1.5 Openness

> *“ Openness to Experience includes active imagination, aesthetic sensitivity, attentiveness to inner feelings, a preference for variety, intellectual curiosity and independence of judgement.*
>
> *People scoring high on Openness tend to be unconventional, willing to question authority and prepared to entertain new ethical, social and political ideas. Open individuals are curious about both inner and outer worlds, and their lives are experientially richer. They are willing to entertain novel ideas and unconventional values, and they experience both positive and negative emotions more keenly than do closed individuals. ”*
>
> – [RC03, p. 69]

The journalist and merchant are likely candidates for openness since they both rely on this trait in their professions in order to seek out deals, stories and opportunities that others miss. The openness trait affects the characters inquisitiveness, they will ask more and about things that are outside the current conversation and they will be less strictly goal oriented. These characters are useful when engaging the player and guiding the drama towards the drama goals since they are less rigidly adherent to specific character goals.

> *“ People scoring low on Openness tend to be conventional in behaviour and conservative in outlook. They prefer the familiar to the novel, and their emotional responses are somewhat muted. ”*
>
> – [RC03, p. 69]

Officers, farmers and religious extremists are low in openness. Officers because their success in the profession is directly correlated with their ability to follow orders and fall in line. Farmers need to be dependable and reliable all year round to tend the farm and manage the family. Religious extremists that blindly follow preachers and old doctrine that cannot be questioned are the very definition of a closed mind. Although, unlike officers and farmers they are likely emotionally unstable.

#### 4.5.1.6 Agreeableness

> *“ An agreeable person is fundamentally altruistic, sympathetic to others and eager to help them, and in return believes that others will be equally helpful. ... found that Agreeableness is related to training success. The co-operative nature of agreeable individuals may lead to success in occupations where teamwork and customer service are relevant ”*
>
> – [RC03, p. 69]

The merchant and journalist are agreeable or else they would not be competent in their profession, which relies on getting along with people. Officers are certainly not disagreeable but more to the middle since they rely on teamwork but are also prepared to fight and will not give in easily.

> *“ The disagreeable/antagonistic person is egocentric, sceptical of others’ intentions, and competitive rather than co-operative. ”*
>
> – [RC03, p. 69]

The smuggler and terrorist are low in agreeableness, which affects how trusting and willing to cooperate the characters are, and how quickly they become threatening. Agreeable people are unlikely to oppose society as strongly as a smuggler, and a terrorist.

#### 4.5.1.7 Conscientiousness

> *“ Conscientiousness refers to self-control and the active process of planning, organising and carrying out tasks . The conscientious person is purposeful, strong-willed and determined. Conscientiousness is manifested in achievement orientation (hardworking and persistent), dependability (responsible and careful) and orderliness (planful and organised).*
>
> *On the negative side, high Conscientiousness may lead to annoying fastidiousness, compulsive neatness or workaholic behaviour.*
>
> *... conceptual relationship between Conscientiousness and integrity. Furthermore, autonomy and goal setting influence the relationship between Conscientiousness and job performance ”*
>
> – [RC03, p. 69]

Officers are high in conscientiousness, a necessary quality for an army career, and farmers that are continually working and managing to provide for their families. Merchants are also high in conscientiousness in order to manage their finance that can be a balancing act, and they require a certain degree of trust from their customers. The customer needs to believe that they will not short-change them or sell them a false product or they will not come back again.

They will have higher rewards for truthfulness and trustworthiness when reasoning.

> *“*
>
> *Low scorers may not necessarily lack moral principles, but they are less exacting in applying them.*
>
> *”*
>
> – [RC03, p. 69]

The unscrupulous journalist that is willing to do almost anything for a good story could be here, and a smuggler and a terrorist are lacking in conscientiousness, or they would not be able to carry out their intentions.

### 4.5.2 Knowledge base

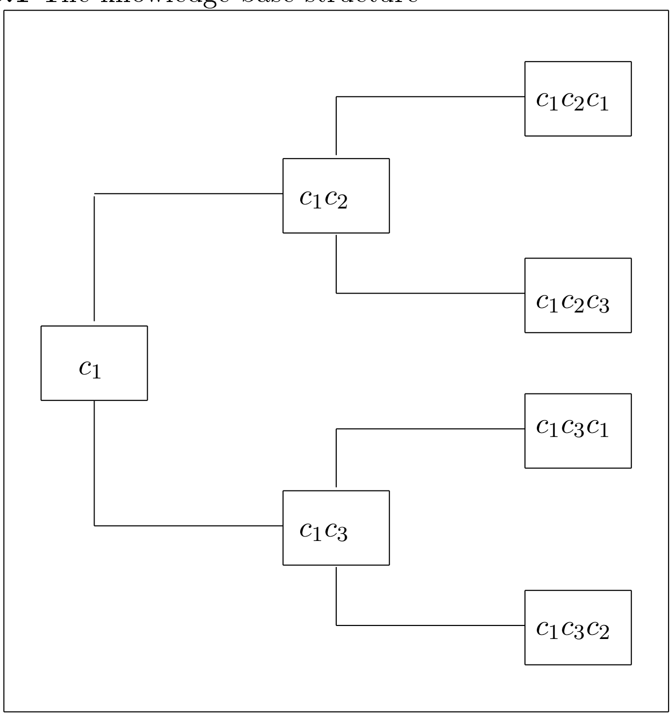
When the director assigns roles to actors, it also sends the actors their initial knowledge base in a list of vertices and their evidence.

The characters will get at least the information in their submodels, i.e. the officer gets all the officer submodel knowledge; see Figure 4.4.2. They will also get information relevant to their profession; for example, the farmer gets all the farms' knowledge.

#### 4.5.2.1 Structure

We introduced the knowledge base structure as part of the RDE in Section 2.4.2. We described it with a diagram of the knowledge base structure necessary for the GT calculation; we display it here again for ease, see Figure 4.5.1.

**Figure 4.5.1** The knowledge base structure



As seen in Figure 4.5.1, the character has three views:

1. $c_1$, their character view is what they know, and they will only enter confirmed data there that comes from someone they trust or has evidence. For example, the officer trusts the OC, and stores all information from the OC in the first view.
2. $c_1c_2$, the second view, their view of other characters, they store information that they find believable. The traits come into this decision; more trusting characters, for example, characters high in agreeableness, will more readily store information here than others. Next comes high openness, followed by extroverts, but neurotic individuals will be very resistant to keeping the information in this layer.
3. $c_1c_2c_1$, the third view is what the character believes that the opponent believes about the character. Anything that the opponent tells the character and the character does not refuse or challenge is stored here. Information in the third layer is also easily replaced if something new comes up. This layer is effectively a memory of what the opponent has said in the past.

#### 4.5.2.2 Technical considerations

In our Afghan Poppy example, we use SpatialOS technology [Imp21a], a highly optimised distributed system for MMOGs. In SpatialOS, as in most online distributed systems, the processes run fast iterations over several agents. In SpatialOS the processes are called workers, and in our default proof of concept, we have the following worker setup:

- Director worker that runs the director tasks.
- Plot worker for the DPGE.
- A NPC worker that runs all the actors per iteration cycle.
- A player worker that runs the player tasks.

Games are run by constantly iterating over all agents that need to react or move as part of the game. The SpatialOS workers who run the agents are stateless; each worker iterates over multiple agents, common in distributed systems. The workers are stateless; therefore, everything needs to be read fresh from a persistent database and inserted or updated at the end of the iteration. In SpatialOS, this database is object-oriented and in the form of entities that the workers communicate via messages. For every iteration, the actors read all data they require from the database and send it back at the end as a list of [submodel, vertex, state].

When they reason about the world, they instantiate the BNs that they have for the relevant character. Since the workers are stateless, the cost of

 121

reading and writing back, all relevant information is added to the time it takes to decide since it is an effective factor in replying sufficiently fast to a player.

#### 4.5.2.3 Initially populate the database

When the actors receive their initial knowledge, they enter all the evidence into their own confirmed knowledge base. The BN represents their character's view of the world, the leftmost $c_1$ vertex in Figure 4.5.1. Next, they update the network and read any additional propagated evidence.

They require an understanding of what $c_1$ believes that the other characters think for the MAIDs. The actors can reason about it by looking at any instantiated vertices that are in role subnets of other characters. For example, if the UK destroyed the farmer's crop, then the vertex for the farmer's crop destroyed has propagated evidence. The actors, as seen in Figure 4.5.1 replicates this evidence to:

They require an understanding of what $c_1$ believes that the other characters think for the MAIDs. The actors can reason about it by looking at any instantiated vertices in the role subnets of other characters. For example, if the UK destroyed the farmer's crop, then the vertex for the farmer's crop destroyed has propagated evidence. The actors, as seen in Figure 4.5.1, replicates this evidence to:

1. $c_1c_2$ their character's view of the opponent, what the officer thinks the farmer knows.
2. $c_1c_2c_1$, the third view, what the officer believes that the farmer believes that the officer knows.

#### 4.5.2.4 Verifying information

Before adding information to their database, the actors $a_1$ must evaluate how believable their character $c_2$ thinks it is. Their reasoning uses a small BN with relevant information extracted from the other BNs.

**Example 4.5.1** *The OC says to the officer that the farmer is growing poppies. The incoming information is vertex growing poppies, state true,* $5_0$ *see Figure 4.5.2*

First, we define the conditional probability for the vertices in Figure 4.5.2, and then we explain how to evaluate incoming data.

The $TRUTHFUL$ vertex has the conditional probability defined in Table 4.5.2, it essentially informs how much the existing and incoming information matches.

The $TRUSTED\_DERIVED$ table, see Table 4.5.3, aggregates how truthful and trusted the information is to decide which information to trust.

 122

**Figure 4.5.2** $a_1$ is evaluating how $c_1$ expects that $c_2$ will react to $5_0$, displayed in the GeNIe UI [DSL07a].

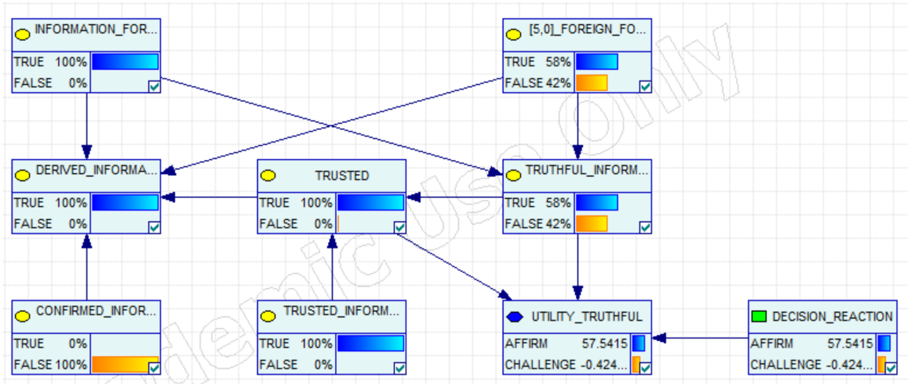



**Table 4.5.2** Conditional probability of the $TRUTHFUL$ vertex in Figure 4.5.2. TRUE(T), FALSE(F)

| $INFORM$ | $T$ | | $F$ | |
|---|---|---|---|---|
| $EXISTING$ | $T$ | $F$ | $T$ | $F$ |
| $TRUE$ | 1 | 0 | 0 | 1 |
| $FALSE$ | 0 | 1 | 1 | 0 |

**Table 4.5.3** Conditional probability of the $TRUSTED\_DERIVED$ vertex in Figure 4.5.2. TRUE(T), FALSE(F)

| $TRUTHFUL$ | $T$ | | $F$ | |
|---|---|---|---|---|
| $TRUSTED$ | $T$ | $F$ | $T$ | $F$ |
| $TRUE$ | 1 | 1 | 0.99 | 0 |
| $FALSE$ | 0 | 0 | 0.01 | 1 |

**Table 4.5.4** Conditional probability of the $DERIVED$ vertex, in Figure 4.5.2, the table is divided in half to fit it on the page.

The first half

| $TRUSTED$ | $T$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $INFORM$ | $T$ | | | | $F$ | | | |
| $EXISTS$ | $T$ | | $F$ | | $T$ | | $F$ | |
| $CONFIRM$ | $T$ | $F$ | $T$ | $F$ | $T$ | $F$ | $T$ | $F$ |
| $TRUE$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| $FALSE$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

The second half

| $TRUSTED$ | $F$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $INFORM$ | $T$ | | | | $F$ | | | |
| $EXISTS$ | $T$ | | $F$ | | $T$ | | $F$ | |
| $CONFIRM$ | $T$ | $F$ | $T$ | $F$ | $T$ | $F$ | $T$ | $F$ |
| $TRUE$ | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| $FALSE$ | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |

The $DERIVED$ table aggregates all of this; see Table 4.5.4. $c_1$ believes information that comes from a trusted source or is truthful. The confirmed information ensures that the incoming data will result if there is no prior verified information. If there is prior confirmed information, then the preceding statement is returned.

The utility has both truthful and trusted as parents to give utility value to each independently. For example, a neurotic character is less trusting and can have distorted views. At the same time, an agreeable individual may be more willing to accept unreliable data than a disagreeable person; see Table 4.5.5. The utility rewards likely information and a trusted source but punishes untrusted source or unlikely information.

**Table 4.5.5** The initial value of the utility vertex in Figure 4.5.2. AFFIRM(A), CHALLENGE(C), TRUE(T), FALSE(F)

| $TRUTHFUL$ | $T$ | | | | $F$ | | | |
|---|---|---|---|---|---|---|---|---|
| $TRUSTED$ | $T$ | | $F$ | | $T$ | | $F$ | |
| $DECISION$ | $A$ | $C$ | $A$ | $C$ | $A$ | $C$ | $A$ | $C$ |
| Value | 100 | 0 | 0 | 0 | 0 | 0 | 0 | -100 |

1. We enter existing data into the top-right vertex, which is named by the vertex id and state index $5_0$.
2. New information is entered into the top-left vertex $INFORMATION\_$.
3. The bottom left $CONFIRMED\_INFO$ vertex is whether the existing information was confirmed. In this example, it is $FALSE$ because the officer has no prior knowledge.
4. We populate $TRUSTED\_INFORM$ vertex with the characters trust in the opponent; in our example, the officer trusts the OC.

### 4.5.3 Selecting actions

When an actor has a goal that is currently relevant, they will decide to act. They evaluate the relevance of the goal depending on whether they are in the same location as the player and whether their knowledge base has the required knowledge. We use Example 4.5.2 to demonstrate how to select actions.

**Example 4.5.2 (Vertices 4 and 5)**

> *In act one in the Afghan Poppy drama, the OC $a_1c_1$ gives the officer $a_2c_2$ information about a Poppy farmer, see Section 4.2, which includes all the information that the OC has in the* $FOREIGN\_FORCE\_0$

*submodel, see Figure 4.4.2. In this example, we use we use vertices 4 and 5 to see their initial state in Figure 4.5.3, and conditional probability in Table 4.5.6. The vertices represent the following knowledge:*

1. $4_0$, *the UK has destroyed a Poppy crop for this farmer.*
2. $4_1$, *the UK has not destroyed the crop.*
3. $5_0$, *the UK knows that the farmer is growing poppies.*
4. $5_1$, *the UK does not know if the farmer is growing poppies.*

**Table 4.5.6** The conditional probability distribution for both 4 and 5 is identical

| parent | $s_0$ | $s_1$ |
|---|---|---|
| $s_0$ | 0.7 | 0 |
| $s_1$ | 0.3 | 1 |

The actor $a_1$ filters out unviable actions that have a lower utility than the initial state, and, therefore, would not improve $c_1$'s situation, as follows:

1. $a_1$ creates an action for all the states in vertices 4 and 5: $[4_0, 4_1, 5_0, 5_1]$.
2. $a_1$ assigns a utility of 100 to all of the information goals; for vertices 4 and 5, they are: $[4_1, 5_0]$.
3. $a_1$ assigns a utility of 1 to all the instantiated states as a reward for truthfulness; for vertices 4 and 5, they are: $[4_1, 5_0]$.
4. $a_1$ evaluates the EU(no action) of the neutral state, the uninstantiated state, and assigns that as the minimum required value.

$a_1$ finds the initial max by calculating the EU as previously introduced, see Definition 2.3.12, and recapped here for this simple case, see Definition 4.5.1. Where for the current state $s_i$; where N is the number of utilities, V is the number of chance vertices, and e is evidence.

**Definition 4.5.1 (The EU function adapted to one chance vertex per utility)**

$$EU(s_i) = \sum_{n=1}^{N} u(V)P(V|s_i, e)$$

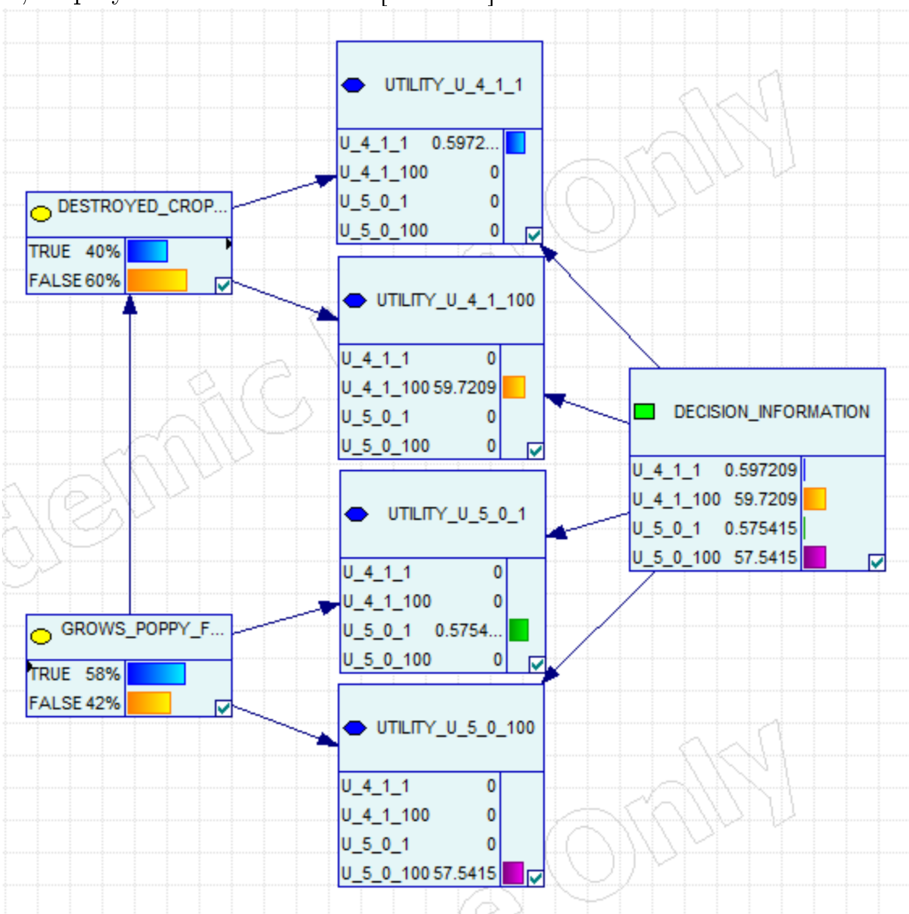
See Example 4.5.3 on how to calculate EU for the initial state

 125

**Example 4.5.3 (Calculating EU(no action) with no state instantiated. )**

| | | | | |
|---|---|---|---|---|
| *EU(no action)* | = | *0.60* | * | *1* |
| | + | *0.60* | * | *100* |
| | + | *0.58* | * | *1* |
| | + | *0.58* | * | *100* |
| *EU(no action)* | = | *0.60* | | |
| | + | *59.72* | | |
| | + | *0.58* | | |
| | + | *57.54* | | |
| *EU(no action)* | = | *118.44* | | |

**Figure 4.5.3** Utility vertex for evaluating an action in an uninstantiated BN, displayed in the GeNIe UI [DSL07a].

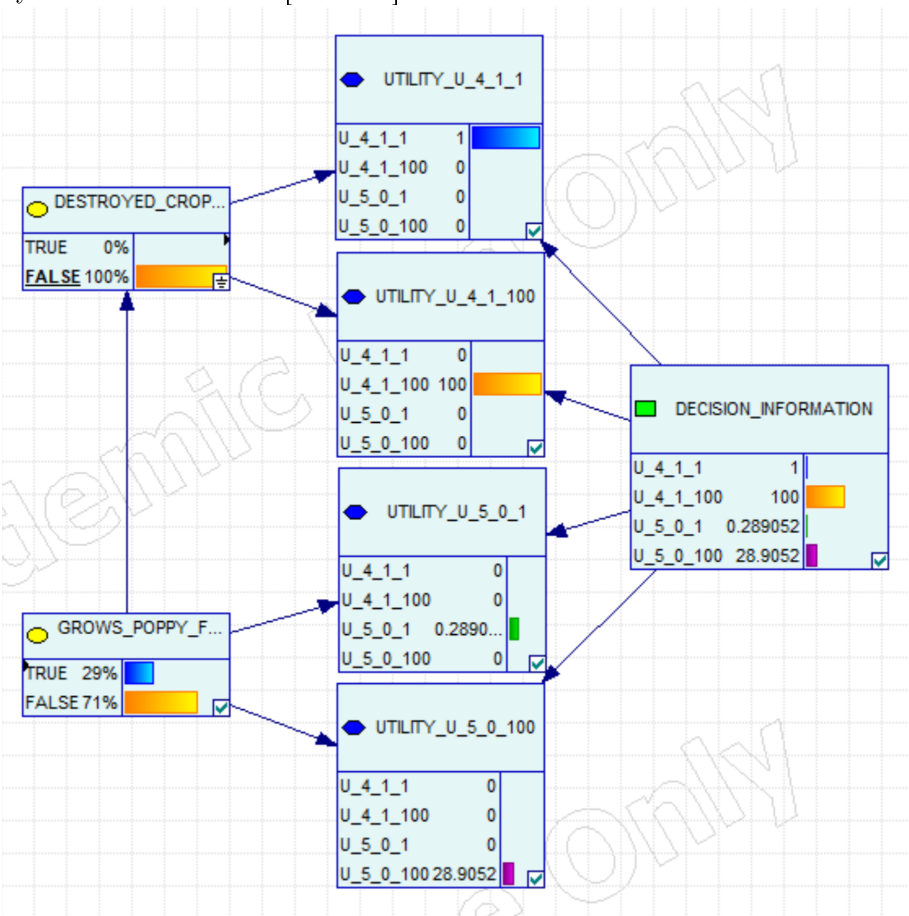


 126

The initial EU(no action) calculated in Example 4.5.3 is 118.44, which $a_1$ sets as the minimum value to filter out all unviable sentences because they have an EU that is less than doing nothing at all. An example of an instantiated state and its utility values is in Figure 4.5.4. $a_1$ now uses the EU(no action) to eliminate unviable actions:

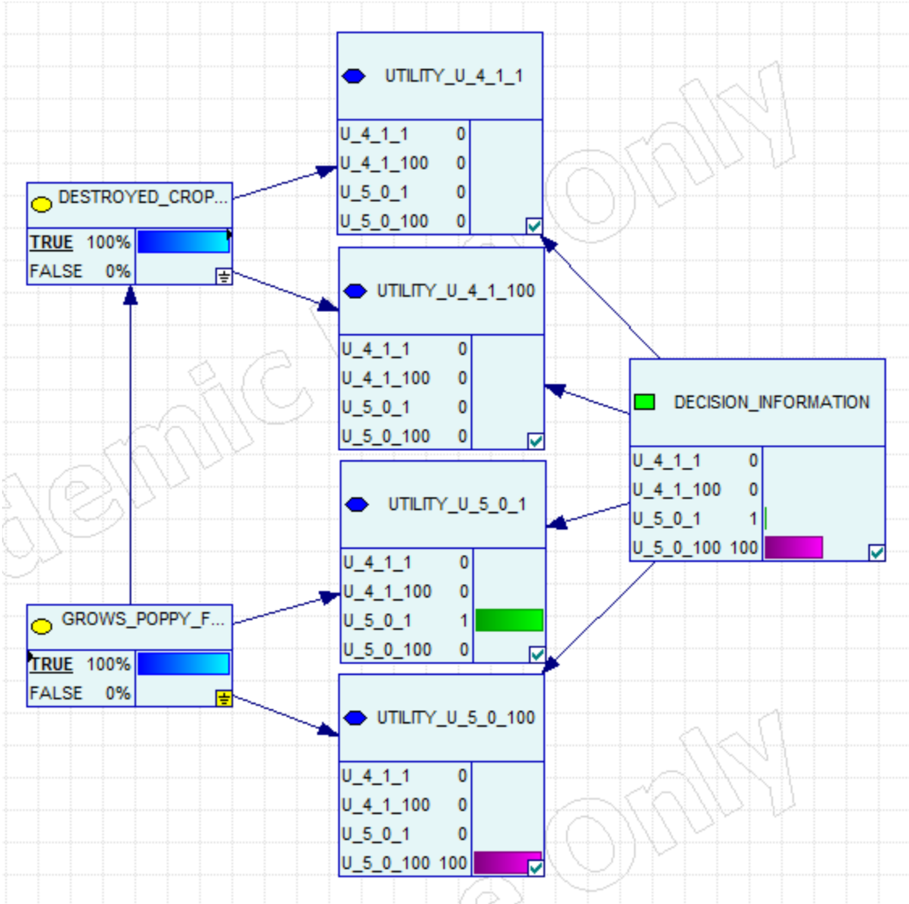
1. $a_1$ instantiates a state, $s_i, e$.
2. $a_1$ calculates the EU of this state, $EU(s_i|s_i, e)$.
3. $a_1$ eliminates the state if the $EU(s_i|s_i, e)$ is less than EU(no action).
4. $a_1$ clears the evidence of $s_i$.
5. $a_1$ repeats from step #1 for all the generated actions.

**Figure 4.5.4** Vertex 4 instantiated to $4_1$ which is its goal condition, displayed in the GeNIe UI [DSL07a].




In Figure 4.5.5 we remove EU($(4_0)$)since it is lower than the initial max. $4_0$ is neither a goal nor supporting a goal; it is, therefore, not yielding any increase in EU.

**Figure 4.5.5** The EU(($4_0$)) is less than EU(no action), displayed in the GeNIe UI [DSL07a].



128

We show the EU for all for states in Example 4.5.4.

**Example 4.5.4 (States with EU less than EU(no action) are eliminated.)**

| | | | | |
|---|---|---|---|---|
| $DESTROYED\_CROP$ | $EU(4_0)$ | *101.00* | $< 118.44$ | |
| $DESTROYED\_CROP$ | $EU(4_1)$ | *130.19* | $> 118.44$ | *Included* |
| $GROWS\_POPPY$ | $EU(5_0)$ | *131.30* | $> 118.44$ | *Included* |
| $GROWS\_POPPY$ | $EU(5_1)$ | *101.00* | $< 118.44$ | |

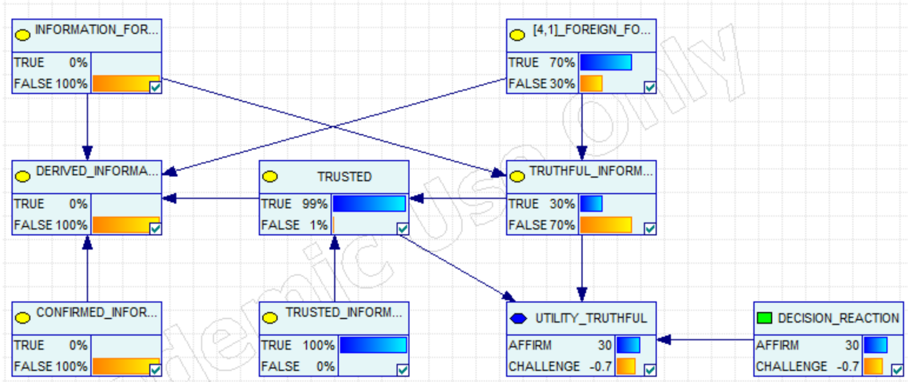
 129

### 4.5.4 Finding an optimal action

Actor $a_1$ has found a set of possible actions in the selection process in Section 4.5.3, and needs to evaluate how the opponent will respond. We continue with Example 4.5.2. Now $a_1$ evaluates each action for the opponents reaction, whether $c_2$ will believe $c_1$. In Section 4.5.2.4, we explained how information is evaluated before being entered into a characters knowledge base. We now show how $a_1$ uses this evaluation to evaluate how $c_2$ will react to $4_1$, see Figure 4.5.6.

and needs to evaluate how the opponent will respond. We continue with Example 4.5.2. Now, $a_1$ considers each action for the opponent's reaction, whether $c_2$ will believe $c_1$. In Section 4.5.2.4, we explained how to evaluate information before storing it in the characters knowledge base. We now show how $a_1$ uses this evaluation to determine how $c_2$ will react to $4_1$; see Figure 4.5.6.

**Figure 4.5.6** $a_1$ evaluating how $c_1$ expects that $c_2$ will react to $4_1$, given that $c_2$ first received $5_0$, displayed in the GeNIe UI [DSL07a].




1. $a_1$ takes the $\mathrm{EU}(4_1)$ values for the opponent's response.
2. $a_1$ adds them to the current EU, which is 130.19; see Example 4.5.5
3. $a_1$ repeats this for $5_0$.
4. $a_1$ collects the set of sentences with the maximum total EU as a list of vertex and state, in this case: $[5_0]$

**Example 4.5.5 (Stage two, EU is calculated for each state and states with reduced EU are eliminated.)**

| | | |
|---|---|---|
| $EU(4_1)$ | $=$ | *130.19* |
| | $+$ | *30* |
| | $+$ | *-0.7* |
| $EU(4_1)$ | $=$ | *159.49* |

The actor $a_1$ now updates $c_1$'s knowledge base as follows.

1. $a_1$ adds $[5_0]$ to $c_1c_2$'s knowledge base to reflect what $c_1$ believes that $c_2$ knows.
2. $a_1$ adds $[5_0]$ to $c_1c_2c_1$'s knowledge base to reflect what $c_1$ believes that $c_2$ believes that $c_1$ knows.
3. $a_1$ instantiates the BN with the evidence that was added to $c_1c_2$'s knowledge base, updates the BN and gathers any propagated evidence to $c_1c_2$'s knowledge base.
4. $a_1$ instantiates the BN with the evidence that was added to $c_1c_2c_1$'s knowledge base, updates the BN and gathers any propagated evidence to $c_1c_2c_1$'s knowledge base.
5. $a_1$ sends $[5_0]$ to $c_2$.

Step 3 and 4 are essential for subsequent reasoning. In some cases, new evidence will cause related vertices to become instantiated. For Example, $[4_0]$ will cause $[5_0]$ to be instantiated because the UK cannot destroy the Poppy crop if they do not know it. For the actor to evaluate if there are relevant goals, the actor needs an updated knowledge base. For Example, when the officer has all the information required, the officer can report back to the OC. The actor knows this by comparing its goals checking to the knowledge base:

The officer, $c_2$, receives $[5_0]$, and the actor, $a_2$:

1. Evaluates whether $c_2$ should trust and believe the information.
2. Receives identical result as a1 since they have identical assumptions.
3. Trusts the OC and confirms the information
4. Adds it to all three views of the knowledge base

After $c_2$ processes $5_0$, as described above, the knowledge base is updated and as a result the $\mathrm{P}(4_1)$ is reduced because now it is in fact the $\mathrm{P}(4_1|5_0)$. As as seen in Figure 4.5.6, the $\mathrm{P}(4_1)$ has fallen from 60%, shown in Figure 4.5.3 to 30%, making it less believable.

#### 4.5.4.1 Asking a question

We will use Example 4.5.6, to explain how to ask a question.

**Example 4.5.6 (Giving aid to the farmer)**

> *In act II, the officer starts to carry out his orders from act I; to visit a nearby farmer and determine whether the farmer is growing poppies. The army is trying to reduce Opium farming and offers aid to those who are not growing poppies or are willing to stop. The farmer has an incentive to lie about his Opium farming to gain support.*

Actor $a_2$'s goal is to determine whether the farmer $c_3$ is growing poppies. From this goal the $a_2$ can generate relevant questions by fining the parents and grand parents of the aid vertex. Based on the aid vertex ancestors, $a_2$ has three things to ask about; is $c_3$ growing poppies, what is he growing on his farm and will he grow Poppy in the future. The officer asks about the Poppy, whether he is growing poppies and what he is growing on each farm.

Actor $a_2$'s goal is to determine whether the farmer $c_3$ is growing poppies. From this goal, the $a_2$ can generate relevant questions by finding the aid vertex's parents and grandparents, which yields three questions for $c_3$, the farmer:

1. Is the farmer growing poppies?
2. What is the farmer growing on each of his farms?
3. Will the farmer grow Poppy in the future?

Extraversion affects how many of the available options the character uses. In our example, the officer has high extraversion and chooses all three questions.

#### 4.5.4.2 Replying to a question

An essential part of drama is conflict and deception; everything is not as it first appears. It is also an essential part of games, to have the player work at solving a problem or a mystery. Therefore, characters do not always tell the truth.

Continuing with Example 4.5.6, we explain how $c_3$ can respond given its knowledge base and that $c_3$ is growing poppies, see Example 4.5.7 for $c_3$'s crop per farm.

**Example 4.5.7 ($c_3$ 's crop on each farm, where $v_0$ = Poppy and $v_1$ is Grain.)**

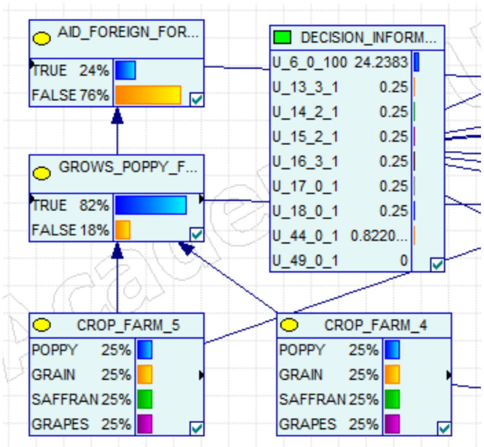
| | |
|---|---|
| $13_1$ | *GRAIN* |
| $14_1$ | *GRAIN* |
| $15_0$ | *POPPY* |
| $16_0$ | *POPPY* |
| $17_0$ | *POPPY* |
| $18_1$ | *GRAIN* |

The $a_3$ starts by selecting viable actions, as described in Section 4.5.3. $a_3$ has the goal of getting aid $v_s = 6_0$ with a value of 100. vertex 44 is whether he is growing poppies and vertex 49 is whether he intends to grow Poppy in the future.

There are 3 actions that have a max, $[13_1, 14_1, 18_1]$, see Example 4.5.8, These are the three farms where $c_3$ is growing grain and not Poppy. He could now send all of these, but he does not have a high extrovert trait and so he picks one of these and sends it.

Initially $c_3$ loses EU by saying that he is growing poppies, see Figure 4.5.8, compared with unset state in Figure 4.5.7, and, therefore, does not select it as a viable action initially. He gains far more from revealing other farms where he grows something else than Poppy, for example Grapes as seen in Figure 4.5.9

**Figure 4.5.7** The EU(no action) for $c_3$ is circa 25, displayed in the GeNIe UI [DSL07a].




**Example 4.5.8 ($a_3$ eliminates any reply that is less than the EU(no action) = 45.315820, $a_3$ considers how $c_2$ will respond and adds it to the EU.)**

**Figure 4.5.8** Initially $c_3$ loses from saying he is growing poppies, EU = 13.25, displayed in the GeNIe UI [DSL07a].

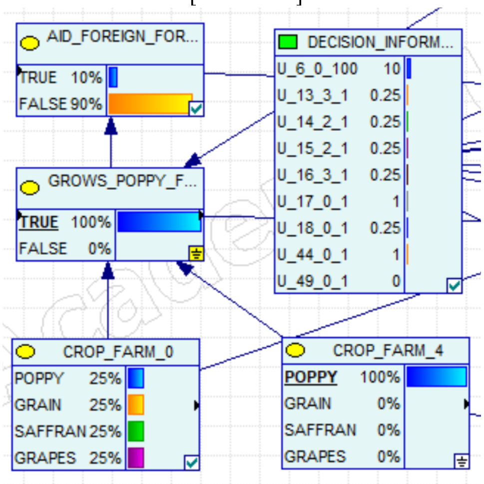
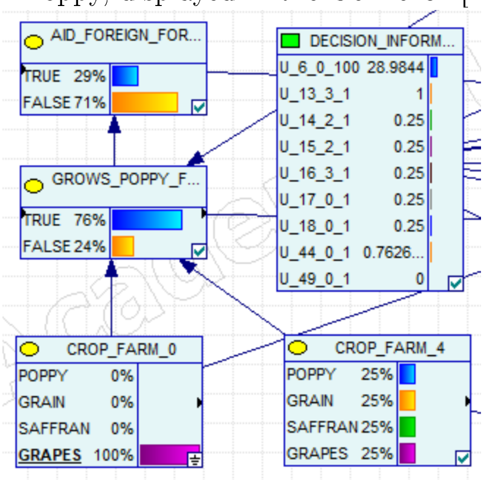



**Figure 4.5.9** He gains far more from revealing other farms where he grows something else than Poppy, displayed in the GeNIe UI [DSL07a].



| *v s* | *Goal* | *Value* | *Included* | *Opponent* | *EU* | *Max Set* |
|---|---|---|---|---|---|---|
| $13_0$ | *0* | *36.95* | | | | |
| $13_1$ | *1* | *48.77* | > 45.32 | *-12.88* | *35.90* | *True* |
| $13_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $13_3$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $14_0$ | *0* | *36.95* | | | | |
| $14_1$ | *1* | *48.77* | > 45.32 | *-12.88* | *35.90* | *True* |
| $14_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $14_3$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $15_0$ | *1* | *37.95* | | | | |
| $15_1$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $15_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $15_3$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $16_0$ | *1* | *37.95* | | | | |
| $16_1$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $16_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $16_3$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $17_0$ | *1* | *37.95* | | | | |
| $17_1$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $17_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $17_3$ | *0* | *36.95* | | | | |
| $18_0$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $18_1$ | *1* | *48.77* | > 45.32 | *-12.88* | *35.90* | *True* |
| $18_2$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $18_3$ | *0* | *47.77* | > 45.32 | *-12.88* | *34.90* | |
| $44_0$ | *1* | *37.31* | | | | |
| $44_1$ | *0* | *82.30* | > 45.32 | *-50.50* | *31.80* | |
| $49_0$ | *1* | *31.44* | | | | |
| $49_1$ | *0* | *68.82* | > 45.32 | *-50.50* | *18.32* | |

$a_3$ sends $14_1$ to the $a_2$, and then adds this answer to $c_3$'s knowledge base as described in Section 4.5.2. When $a_2$ receives the response and evaluates it, then $a_2$ gets a negative evaluation which means that $c_2$ wants to challenge the reply.

To challenge information received $a_2$ needs to find some argument to challenge it with. To do this $a_2$ looks at all parent vertices for any evidence and gathers all vertices that have evidence. The traits, Openness and conscientiousness affect this reasoning, the higher a character is in either trait the larger part of the BN it can search. $a_2$ finds one vertex with evidence, $c_2$ knows that $c_3$ is growing poppies because $c_1$ informed $c_2$ of this in act one. Next $a_2$ starts to go through the same steps as before and calculates the EU(no action), which is 75.000000. Then $a_2$ calculates EU for $44_0$. the goals are the other crop types, see Example 4.5.9. The EU($44_0|44_0$) is 150.018510 which is greater than 76.804277. So $c_2$ will gain by revealing it to the $c_3$. Then $a_2$ evaluates the $c_3$ response which is 73.214233. The final evaluation is, therefore 150.018510.

**Example 4.5.9 ($a_2$'s goals when challenging $c_3$)**

| $v$ $s$ | *Goals* |
|---|---|
| $14_0$ | *100* |
| $14_2$ | *100* |
| $14_3$ | *100* |

$a_2$ has now determined that there is a gain by saying that $c_3$ grows Poppy, so $a_2$ marks the sentence, $14_0$, from $c_3$ with challenge tag and the $44_0$ with an AFFIRM tag.

When $c_3$ receives these, $a_3$ removes the challenged item from the databases. $a_3$ evaluates the $44_0$ and finds it believable. $a_2$ adds it to $c_2$'s knowledge base.

$a_2$ marks this as resolves and sends the director an update, but $a_2$ still has questions regarding what $c_3$ is growing on his farms and whether $c_3$ will continue to grow Poppy. So $a_2$ follows up with questions about what $c_3$ is growing on the farms.

**Example 4.5.10 (Stage two and three after $c_2$ has told him he knows of his Poppy farming, the EU(no action) is 35.61)**

 136

| $v\ s$ | *Goal* | *Value* | *Included* | *Opponent* | *EU* | *Max Set* |
|---|---|---|---|---|---|---|
| $13_0$ | *0* | *35.25* | | | | |
| $13_1$ | *1* | *36.43* | > 35.61 | *-15.59* | *20.84* | |
| $13_2$ | *0* | *35.43* | | | | |
| $13_3$ | *0* | *35.43* | | | | |
| $14_0$ | *0* | *35.25* | | | | |
| $14_1$ | *1* | *36.43* | > 35.61 | *-15.59* | *20.84* | |
| $14_2$ | *0* | *35.43* | | | | |
| $14_3$ | *0* | *35.43* | | | | |
| $15_0$ | *1* | *36.25* | > 35.61 | *-4.73* | *31.52* | *True* |
| $15_1$ | *0* | *35.33* | | | | |
| $15_2$ | *0* | *35.33* | | | | |
| $15_3$ | *0* | *35.33* | | | | |
| $16_0$ | *1* | *36.25* | > 35.61 | *-4.73* | *31.52* | *True* |
| $16_1$ | *0* | *35.33* | | | | |
| $16_2$ | *0* | *35.33* | | | | |
| $16_3$ | *0* | *35.33* | | | | |
| $17_0$ | *1* | *36.25* | > 35.61 | *-4.73* | *31.52* | *True* |
| $17_1$ | *0* | *35.33* | | | | |
| $17_2$ | *0* | *35.33* | | | | |
| $17_3$ | *0* | *35.33* | | | | |
| $18_0$ | *0* | *35.25* | | | | |
| $18_1$ | *1* | *36.43* | > 35.61 | *-15.59* | *20.84* | |
| $18_2$ | *0* | *35.43* | | | | |
| $18_3$ | *0* | *35.43* | | | | |

The second time that $c_2$ asks what crop $c_3$ is growing, $c_3$ has $44_0$ (Growing Poppy True) instantiated in all three knowledge models; $c_3$, $c_3c_2$, and $c_3c_2c_3$. When $a_3$ starts the reasoning the numbers are different from before, see Example 4.5.10. Now it has become advantageous to admit that he is growing poppies to be truthful. The connection between the crops and the AID vertices is a serial connection and, therefore, evidence on the grows Poppy vertex, see Figure 4.5.10, results in a D-separation between the crop vertices and the AID. Evidence give to the crop vertices are therefore no longer affecting the AID utility. In fact, now there is a somewhat higher EU for admitting to a Poppy farm because of the truthful goal. Furthermore, there is a convergent connection between the crops. Before any evidence on grows Poppy true, then evidence to one crop did not affect the other crops, but after the evidence is entered then evidence to on ecrop can affect the others. For example, evidence to Saffran wil increase the probability off Poppy in other vertices since it must be true that one of the crops is Poppy, see Figure 4.5.11. Conversely if the grows Poppy vertex is instantiated to false then there is

 **137**

no longer any reward for saying Poppy, since it cannot be true that any of the crops is Poppy. In fact it is not possible to instantiate it, see see Figure 4.5.12.

**Figure 4.5.10** Instantiateing a state on vertex 44 will cause a D-separation, displayed in the GeNIe UI [DSL07a].

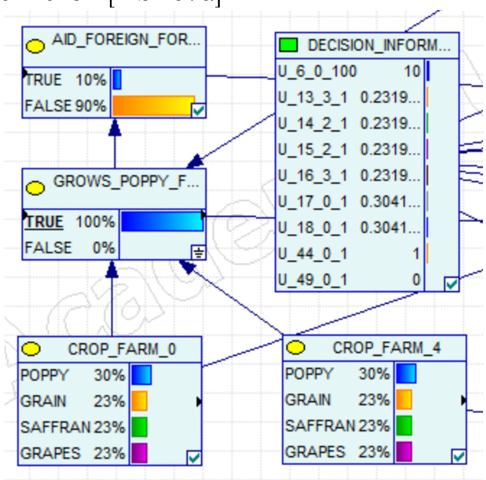
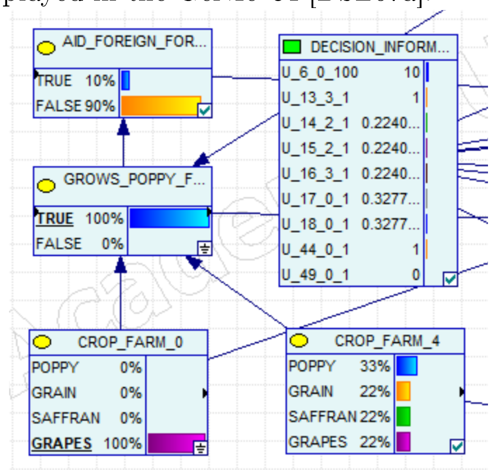



**Figure 4.5.11** Setting crop as Saffran will affect the belief of what is growing in other farms, displayed in the GeNIe UI [DSL07a].




#### 4.5.4.3 Deception

Another scenario could evolve if $c_2$ does not know any information to use to challenge $c_3$ with. $c_2$ will then affirm the knowledge to $c_3$, and add it to the $c_2c_3$, and $c_2c_3c_2$ knowledge bases since he has no reason not too. The actor will not add it to the confirmed knowledge, $c_2$. $c_2$ will start to believe $c_3$ more and more as the knowledge base is increasingly populated with evidence from

 138

**Figure 4.5.12** Saying that the crop is Poppy is no longer possible because $44_1$ is instantiated, displayed in the GeNIe UI [DSL07a].

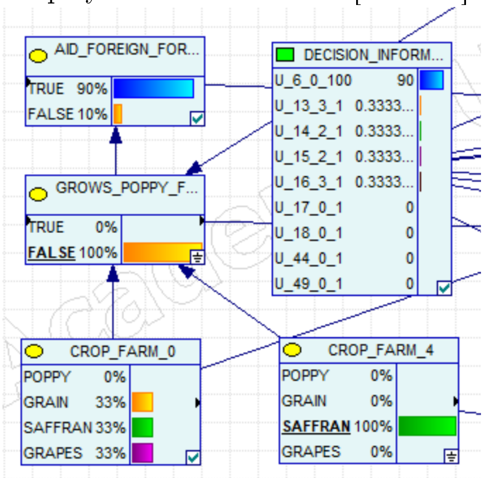
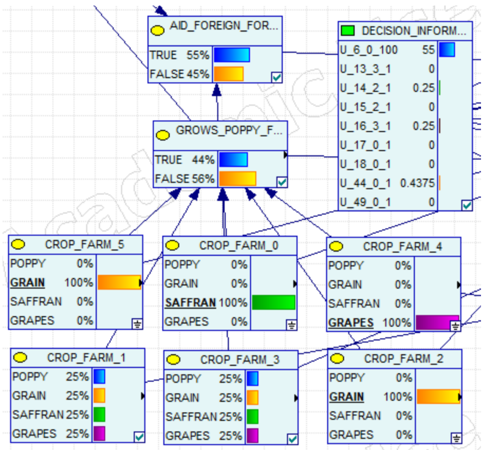



**Figure 4.5.13** By choosing to talk about only non Poppy crops, $c_3$ can make it seem unlikely that he is growing poppies, displayed in the GeNIe UI [DSL07a].

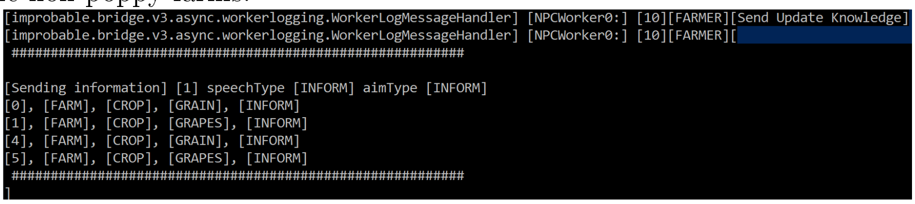


$c_3$. $c_3$ doesn't even need to lie, if he chooses to only discuss the non Poppy crops then the probability of growing poppies vertex changes from probably true to probably false, see Figure 4.5.13. This continues until saying "I'm not growing poppies" becomes the optimal sentence and $c_2$ has no reason to challenge it. For example, in Figure 4.5.14 is a screenshot of a test run where $a_3$ decides to deceive $c_2$. In this particular run $c_3$ has the crops in Example 4.5.11.

**Example 4.5.11** **($c_3$ 's crop on each farm when deceiving $c_2$.)**

| | |
|---|---|
| $13_1$ | *GRAIN* |
| $14_3$ | *GRAPES* |
| $15_0$ | *POPPY* |
| $16_0$ | *POPPY* |
| $17_1$ | *GRAIN* |
| $18_3$ | *GRAPES* |

**Figure 4.5.14** A screenshot of the console showing the farmer informing of the non poppy farms.

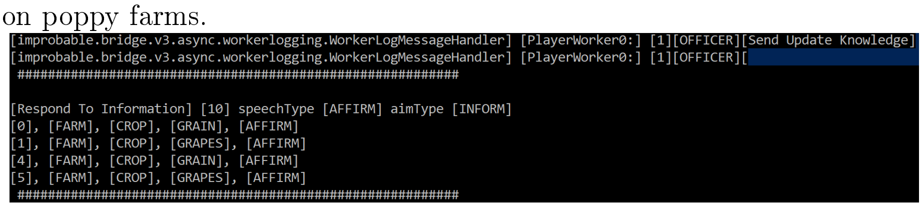


```
[improbable.bridge.v3.async.workerlogging.WorkerLogMessageHandler] [NPCWorker0:] [10][FARMER][Send Update Knowledge]
[improbable.bridge.v3.async.workerlogging.WorkerLogMessageHandler] [NPCWorker0:] [10][FARMER][
##########################################################

[Sending information] [1] speechType [INFORM] aimType [INFORM]
[0], [FARM], [CROP], [GRAIN], [INFORM]
[1], [FARM], [CROP], [GRAPES], [INFORM]
[4], [FARM], [CROP], [GRAIN], [INFORM]
[5], [FARM], [CROP], [GRAPES], [INFORM]
##########################################################
]
```

$a_2$ accepts those statements by affirming them and adds them to in Figure 4.5.15

**Figure 4.5.15** A screenshot of the console showing the officer affirming the non poppy farms.



```
[improbable.bridge.v3.async.workerlogging.WorkerLogMessageHandler] [PlayerWorker0:] [1][OFFICER][Send Update Knowledge]
[improbable.bridge.v3.async.workerlogging.WorkerLogMessageHandler] [PlayerWorker0:] [1][OFFICER][
##########################################################

[Respond To Information] [10] speechType [AFFIRM] aimType [INFORM]
[0], [FARM], [CROP], [GRAIN], [AFFIRM]
[1], [FARM], [CROP], [GRAPES], [AFFIRM]
[4], [FARM], [CROP], [GRAIN], [AFFIRM]
[5], [FARM], [CROP], [GRAPES], [AFFIRM]
##########################################################
```

In Figure 4.5.14, see Figure 4.5.15 the numbers to the far left (0, 1, 4, 5) refer to the number of the *FARM* subnet while the vertex identifier in Example 4.5.11 refers to the identity of the vertex within the BN.

#### 4.5.4.4 Giving goals to other actors

When the OC $a_1$ in act I gives the officer $c_2$ orders then $a_2$ receives goals that are assigned to $c_2$ and aimed at the farmer $c_3$. $a_2$ first needs to translate these into goals for $c_2$. The goal from $a_1 =$ is sent to the actor as other messages, it has the type $ORDER$ and applies to $FARMER$ and type $GAIN\_INFORMATION$ the sentence part is (subnet number, subnet, vertex, state) = (0, $REASONING$, $AID$, $UNKNOWN$). From this information the $a_2$ understands that it needs to ask $c_3$ questions that will determine the state of the $AID$ vertex. To asks questions $a_2$ generates questions as described previously in Section 4.5.3. $a_2$ then proceeds to ask them as described in Section 4.5.4.1.

This process of creating goals for themselves and ask other actors to create goals is used by the actors whenever they need to know what to do to satisfy a goal.

When $a_2$ creates these goals, it marks the goals with action $ORDER$ and role $OFFICER\_COMMANDING$ to be able to create the goal of informing the officer when the order has been carried out.

#### 4.5.4.5 Response to a Request

In Section 4.5.4.3, $c_2$ can be deceived by $c_3$ and may end by walking away believing that $c_3$ is not growing poppies. If it is a drama goal for $c_2$ to learn that $c_3$ is growing poppies then it could be an issue that $c_2$ is unsuccessful in extracting the information.

We resolve this problem by having the actors reaching out to each other for aid, and we call this the duality of drama and character goals. The actors will always prioritise character goals and stay in character. When they realise that they are acting contrary to a drama goal, then they will ask another actor to carry out the drama goal.

In our example above $a_3$ realises that $c_3$ is contradicting a drama goal when $c_2$ affirms that $c_3$ is not growing poppies, given that the drama goal is for $c_2$ to learn that $c_3$ is growing poppies. $a_3$ then asks another actor to assist. To do this the $a_3$ will send a request similar to the order in Section 4.5.4.4 except that it is marked as $ASSIST$ rather than $ORDER$. The receiving actor then knows to create goals from this message.

$a_3$ will try to pick an actor that would be better able to assist, in our example, someone that already knows of the poppies. For example, a journalist that was asking questions or a merchant that travels the country to sell their merchandise. $a_3$ knows this by looking at their knowledge base look for all the actors they have in their knowledge base whether they believe that they know vertex (growing poppy = true). If they do not find anyone that they believe has this information, then they pick one actor at random. They exclude themselves and their current opponent from the set of possible actors to assist.

## 4.6 Players

The DED manages the players, for the most part, as the actors. The director passes schemas to the players as if they were an actor; it is not aware of any difference when selecting schemas or assigning initial knowledge base. The director does know that the player has a specific role according to an initial configuration. After the initial role assignment, the director has the player in its list of actors and handles the player the same as the actors.

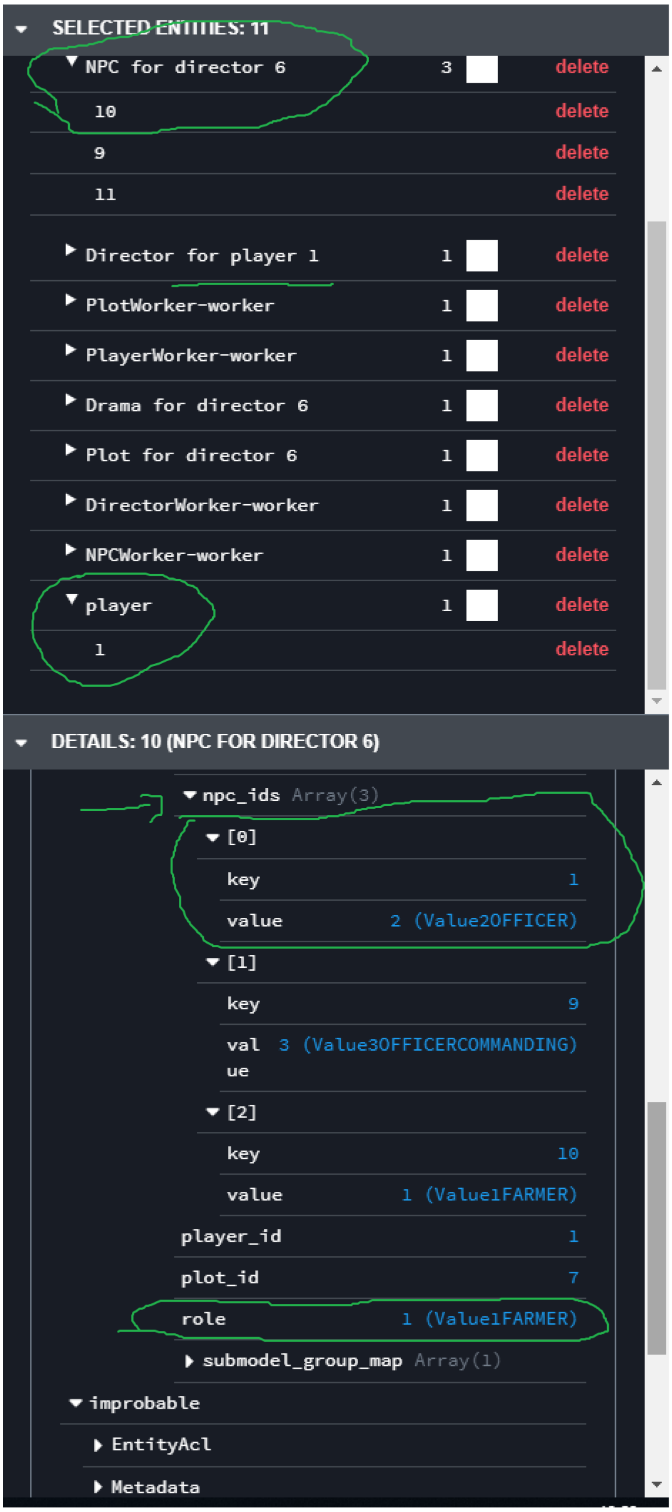
Similarly, for the actors, when they are interacting with other actors, they use a map of identifiers and roles ($map < id, role >$). The player is listed in that map in the same way as the actors are. For example, in Figure 4.6.1, which shows the world view of farmer $NPC_{10}$, the player's listing is identical to other NPCs. At this stage in this particular drama, the farmer only knows of the officer and the OC.

The player receives goals as the other actors, both character and drama goals, and the player also has the same blueprint of the world knowledge that the other actors have. Additionally, like the other actors, the player receives the initial knowledge base from the director. The main difference between player and NPCs is that players make decisions for their character instead of the reasoning algorithm that the NPCs actors use.

An interface for the player has not been built for our SpatialOS implementation, but it has been implemented for other applications that we have tested.

 142

**Figure 4.6.1** The player is listed with the NPCs in the farmer's world view.



## 4.7 Creating a new drama

The DED's novel approach calls far an alternative means of creating a new drama scenario than that of authoring storylines, branching stories of episodes. Instead the main task is to structure the BN, pace the drama, design characters, and to design a user interface.

### 4.7.1 Bayesian Network

As introduced in sections 2.3.1.4, **??**, and 4.5, the BN must be causally structured. It does not call for strict causality, but rather that the direction is from perceived cause to perceived effect. The algorithms rely on this structure for the actors reasoning to produce what can be perceived as coherent. For example, in Section 4.5.4.2 where the officers uses prior knowledge to challenge what the farmer just said. The officer does this by searching related vertices for evidence that will help in proving that the farmer is not telling the truth. He was able to verify that the evidence could help him by evaluating the expected utility. If the BN is not causally constructed but is connected by correlation, this reasoning could go badly. The officer might, for example, say that there is a drought, so he is probably growing poppies. That does not appear coherent to us; we find it to be improbable that poppies grow well in a drought, but in Afghanistan there is a high rate of both drought and poppies growing and there is, therefore, a likely correlation. We realise that they are hardly co-occurring.

Similarly, in Section 4.5.4.3, where the farmer manages to deceive the officer by first talking about the farms where he is growing something else than poppies. The probability that the farmer is growing poppies is gradually reduced in the officers reasoning, with each information that the officer accepts. We rely on the causal structure and the manner in which D-separation works to accomplish this, see Section 2.3.1.4 for introduction to D-separation.

In this thesis, all the BN's are manually structured as a proof of concept. Manual construction is not sustainable and cannot meet the needs of high-end simulations and games. Instead, we are directing our future research efforts to create automated procedures that create the networks from real-world data to modelling training simulations. There is good quality research work to build on, the networks used for medical diagnostic tools are also causally structured, for example, in the words of B. Liew & Co:

> *" The present study is the first to* ***apply BN modelling to understand the causal mechanisms*** *of recovery in whiplash associative disorders. It is anticipated that such analytical methods could increase the precision of treatment in individuals with chronic whiplash associative disorders. "*
>
> – (FA: we added emphasis for clarity) [LSP+19, p. 1]

### 4.7.2 Pacing the drama

As explained in Section 4.3.1, the drama is divided into acts and to move between acts all goals for the act need to be fulfilled. The author can use this to guide the drama to emerge along the dramatic arc. There can be as many acts as the author needs. The author defines each drama goal and for which act and schemas it belongs. This is the authors way of guiding the emergence of the drama and can increase the complexity gradually by increasing the number or difficulty of drama goals and character goals. A schema can only trigger in its designated act. Therefore the author designates schemas to the acts that they should trigger in. In a 5 act drama then introductory schemas are designated tor act 1 and 2. Schemas that build tension are designated to act 3 and 4, and Schemas for the climax, i.e to reveal a murderer to act 4 and the resolution schemas to act 5.

**Roles** Each actor has at least one role that the director assigns at the start, and this role decides what initial knowledge the actor receives. In out Afghan poppy scenario, the main roles are officer, OC and farmer. There are also side-roles: merchant, journalist, smuggler, and terrorist. Each of these roles will need some accompanying characteristics or traits to identify them; for example, the journalist is extrovert and high in openness. There can also be lesser roles assigned on the fly as needed by the director, for example, a brother or a husband in family affairs. Also, in crimes, there can be victim and culprit roles. The merchant could for example, also be a victim of a theft.

**Goals** schemas contain all drama and character goals that the actors and player use—the goals map to specific vertex and state in the BN. To create more exciting drama, it is good to have together drama goals and character goals that conflict and cause the characters to attempt deception and pass their drama goals to other actors. This is what causes more significant conflict and suspense and increases the interestingness [SER16] of the drama.

### 4.7.3 Author speech

similar to constructing the BNs we have manually authored sentences to match the states in the BN.

## 4.8 Summary

The DED architecture is decentralised drama management; comprised of a director agent that guides rather than directs the drama through the deployment of relevant schemas.

In DED the director gives the actors drama goals which inform the actors on how to progress the drama, and character goals for their character play. The director does not know what the actors or players do.

The actors are aware that they are actors playing characters in a drama and are the prima motor of the drama. They are responsible for progressing the plot, engaging the player and responding timely, coherently and IC to the players actions.

This is unlike any other system that has a director type agent.

We are using improv which is applicable for cooperative games because it assumes that the actors are cooperative and aid each other. Improv is not really applicable for non-cooperative games. Using improv the actors take responsibility for the drama and are the prima-motor of the drama management. The actors communicate between them selves to further the drama goals. The actors are aware of the drama and their responsibility to manage the drama. The actors use character and drama goals to understand whether they are furthering the requirements of the drama or just chasing selfish goals. This is coined as the single biggest problem of interactive drama in the overview of the "UK network RIDERS Research in Interactive Drama Environments, Role-Play and Story-telling, running for 36 months from September 2011" [ALW11, p. 1].

The DED has implemented, tested and published in [AK08, AK09b, AK09a, AK10, AK12]

# 5

# Evaluation

In this chapter we first introduce each application that we have implemented to test the DED. The applications are listed below in the temporal order of their implementation:

1. Section 5.1: Murder mystery in Second Life.
2. Section 5.2: Web-based murder mystery.
3. Section 5.3: Mistlethorpe: a web-based murder mystery.
4. Section 5.4: Merchants of Jorvik: a web-based dramatisation of life in York during the Icelandic Sagas.
5. Section **??**: Poppie farming scenario, introduced in Section see Section 4.2 and used as a running scenario throughout Chapter 4

For each of the above applications we describe and discuss the following elements:

1. Development, design and authoring.
2. Emergent interactive drama.
3. Evaluation.

1. Section 3.2 After introducing the systems and our evaluation of each system we compare our drama metrics' scores against the drama metrics that we established in Section 3.2 and compare our results with related systems' scores.
2. Section **??**

## 5.1 Murder mystery in Second Life

SL is a VR where users can create their own avatar and engage in several activities, including:

- Personalise their 3D avatars.

SL is a complex commercial VR that offers an enormous amount of content and freedom of interaction.

### 5.1.1 Development, design and authoring

We used an open source library, called LibOpenMetaverse [HRK$^+$08], to developed a scenario in SL. & social experiences to share or sell in a global marketplace. The platform we use for tests is the Second Life (SL) [Lab03]; a virtual reality, with over 16 million users. Library called LibOpenMetaverse [HRK$^+$08]. The university of York (UofY) paid for an island in SL that we and other within the UofY could use.

Using the `LibOpenMetaverse` library [HRK$^+$08] we programmed actors that logged in to SL as if human users controlled them. The SL VR supplies a lot of content that is free of use by the comunity and we mainly used props from them including most of the gestures that we used. However, we needed to create an animation of the victim dying and a few of the simple gestures like waving hello and adjusting an animation for shooting from a pistol to that of drinking from a glass. We used a number of gestures to indicate emotions, such as anger, happiness, grief and shock. We uploaded all the animations to the repository along with threee videoclips of the actors being interrogated.[1]

1. All the code base is available at our Department's Github account.[2].
2. Additionally, we provide logs of our work from 2008 to 2011 in that repository.[3].

### 5.1.2 Gameplay

i. **Valla:** Penny, was Snorri blackmailing you ?
ii. **Penny:** Snorri was blackmailing me, Snorri knew a lot, I have secrets... they are deeply personal...

[1] `https://github.com/uoy-research/DED/tree/main/SecondLife/SL%20Animation`
[2] `https://github.com/uoy-research/DED/tree/main/SecondLife`
[3] `https://github.com/uoy-research/DED/tree/main/SecondLife/logs`

148

**Figure 5.1.1** Kenneth exposes penny as a swindler, screenshot from Second Life [Lab03].

## 5.2 Web-based murder mystery

### 5.2.1 Development, design and authoring

The web-based murder mystery used the same C# DED as SL; we only changed the interface to a web environment for our user study. We found that running a user study in SL was infeasible due to the difficulties in sourcing and setting up a test environment for remote participants; we wanted a diverse pool of participants for a broad and realistic user evaluation.
Another significant cause was that UofY was cancelling their SL island. We were the island's primary users over approximately two years, and we rarely saw our colleagues there.
Therefore, we developed a simple web interface and deployed our study to the UofY web-servers.
The game had a few modes for different type of activities:

i. **Cause of death:** The player determines cause of death.
ii. **Motive:** The player determines the motive, if any, of each character.
iii. **Examination:** The player investigates the murder scene; see Figure 5.2.1.
iv. **Speech:** The player can talk to the characters in the story;see Figure 5.2.2.

**Figure 5.2.1** The player examined the body in the web game, an image used is painted over for copyright reasons.


### 5.2.2 User interaction

We created a web based mystery game for user evaluation. As the game starts, the player finds a body and can examine the body and determine the cause of death. Each initiated game has a randomly generated cause of death, and the BN is used to generate

 150

**Figure 5.2.2** A screen shot of a dialogue with suspects in the web game, images used are painted over for copyright reasons.

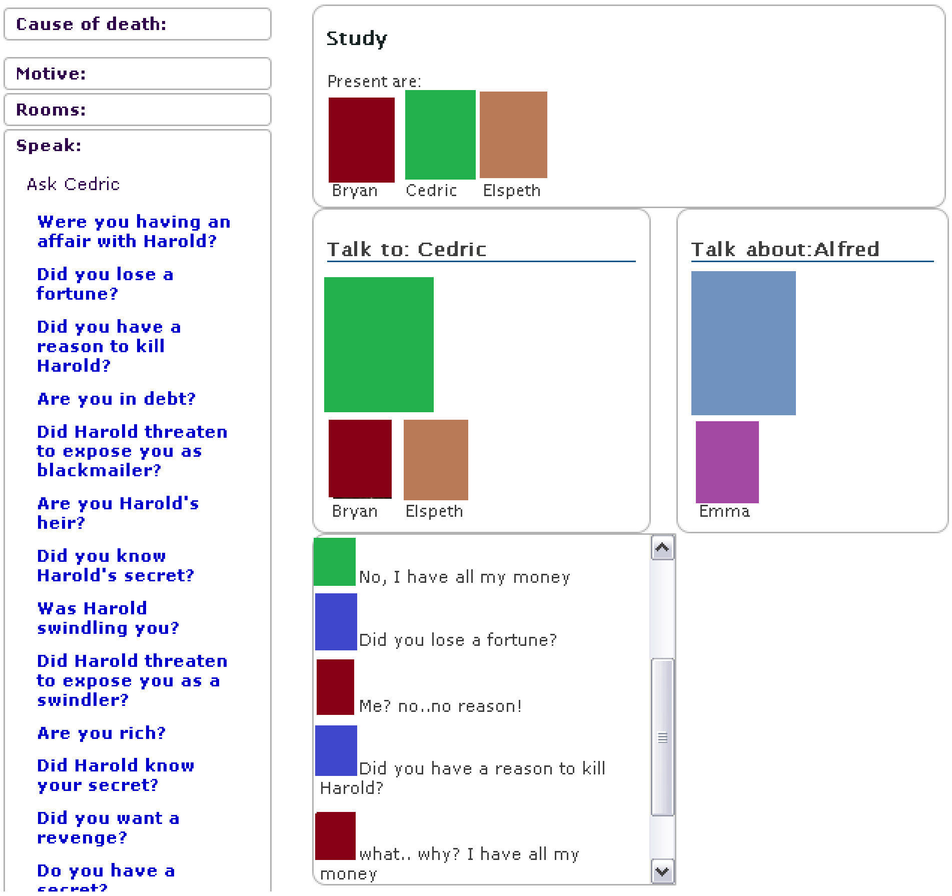
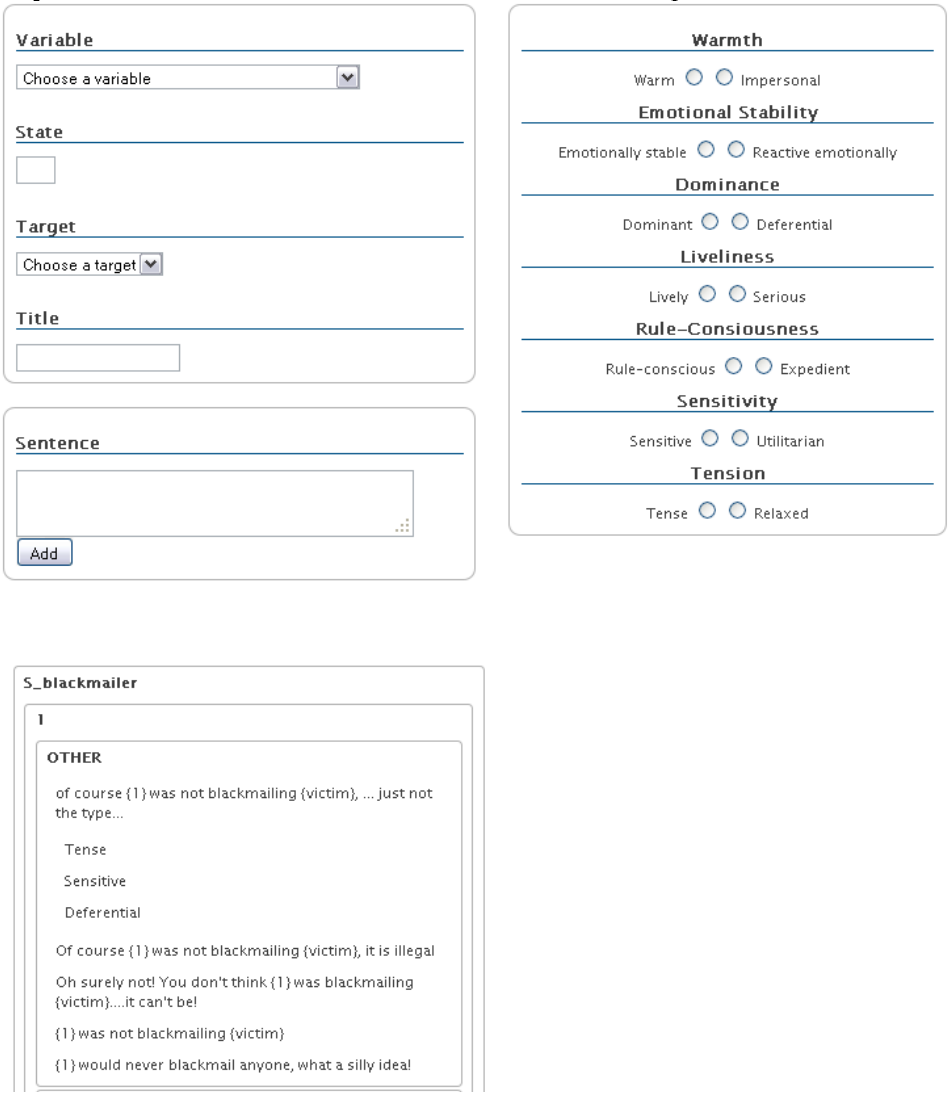



**Table 5.2.1** The 7 Cattell's traits used for the characters

| Low range | High range |
|---:|---|
| reserved | warm |
| reactive | emotionally stable |
| deferential | dominant |
| serious | lively |
| expedient | rule-conscious |
| utilitarian | sensitive |
| relaxed | tense |

 151

**Figure 5.2.3** A screen shot of the sentence authoring tool.



clues on the body to be found by the user that are logically consistent with the cause of death. For example, a strangled victim has marks around the neck, and there may be a strange odour from a poisoning, see Figure **??**.

The player can then explore other rooms and interrogate suspects on their possible motives and attempt to determine which motive, if any, each suspect has. An example dialogue with two suspects shows how they are significantly different personalities, see excerpt below and in Figure **??**

Dialogue (Miss Jane Marple and Mr Bryan Eastley):

i. *Miss Jane Marple* - Did you lose a fortune?
ii. *Mr Bryan Eastley* - what. Why? I have all my money
iii. *Miss Jane Marple* - Did you have a reason to kill Harold?
iv. *Mr Bryan Eastley* - Me? no..no reason!

Dialogue (Miss Jane Marple and Mr Cedric Crackenthorpe):

i. *Miss Jane Marple* - Did you lose a fortune?
ii. *Mr Cedric Crackenthorpe* - No, I have all my money
iii. *Miss Jane Marple* - Did you have a reason to kill Harold?
iv. *Mr Cedric Crackenthorpe* - I assure you, I'm no murderer!

### 5.2.3 Evaluation

#### 5.2.3.1 Computational evaluation

Playtime, A total of 34 participants answered the questionnaire, from the logs we can gain some information in addition to what they answer. We can filter the logs for those that played for more than half an hour and for more than an hour. Some play for a while and then stopped for an hour and came back, and it is dicfficult to say how log those plaeyd in total. So as not to inflate the c=values we filtered out anyone who had spent more than 2 hours as those ar mostly people that didn't play for that entire time.

Therefore we filterd out those that had played between 30 minuted - 2 hours.

i. > 30 minute: 28 of 34
ii. > 1 hour: 9 of 34
iii. > 1 hour and 30 minutes: 7 of 34

Further more the 4 years of logs that we have accumulated and give access to in the code repository demonstrate that the characters answer within a second.

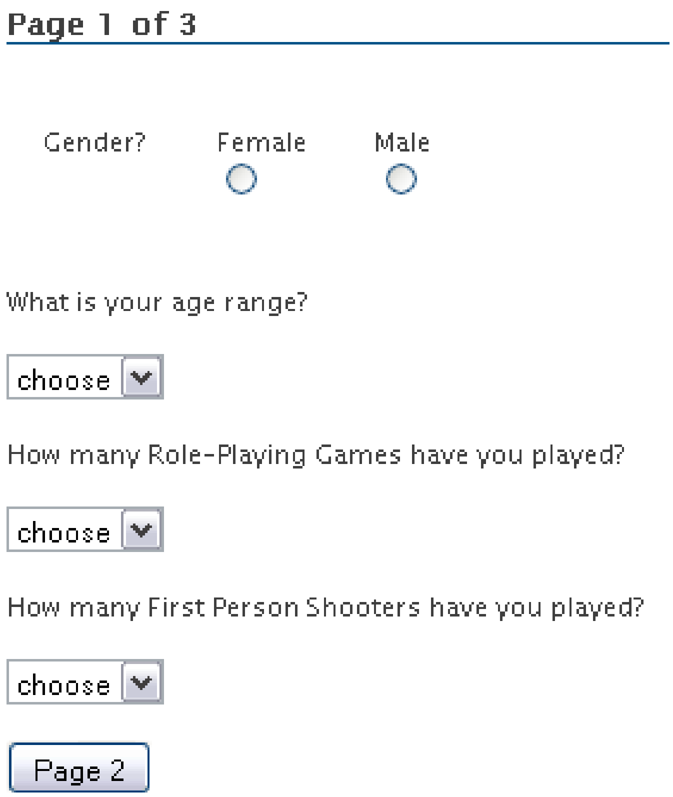
**Table 5.2.2** Participants that played the game between half an hour to two hours

| **Playtime** | **Start time** | **end time** |
|---|---|---|
| 0:49:45.203000 | 2010-10-18 13:22:20 | 2010-10-18 14:12:05.203000 |
| 0:31:25.672000 | 2010-10-18 22:52:44.921000 | 2010-10-18 23:24:10.593000 |
| 0:32:14.484000 | 2010-10-19 22:25:36.687000 | 2010-10-19 22:57:51.171000 |
| 0:32:58.110000 | 2010-10-20 12:34:06.921000 | 2010-10-20 13:07:05.031000 |
| 0:34:17.515000 | 2010-11-03 12:34:51.750000 | 2010-11-03 13:09:09.265000 |
| 1:16:50.594000 | 2010-11-03 23:38:54.359000 | 2010-11-04 00:55:44.953000 |
| 0:30:39.359000 | 2010-11-05 01:40:24.578000 | 2010-11-05 02:11:03.937000 |
| 1:44:54.485000 | 2010-11-05 11:36:57.593000 | 2010-11-05 13:21:52.078000 |
| 1:14:07.625000 | 2010-12-14 13:33:33.093000 | 2010-12-14 14:47:40.718000 |
| 0:51:21.047000 | 2010-12-16 11:38:57.843000 | 2010-12-16 12:30:18.890000 |
| 0:36:53.907000 | 2011-01-24 13:38:05.921000 | 2011-01-24 14:14:59.828000 |
| 0:41:44.672000 | 2011-02-02 23:23:18.164000 | 2011-02-03 00:05:02.836000 |
| 1:52:41.313000 | 2011-02-08 10:55:39.515000 | 2011-02-08 12:48:20.828000 |
| 0:34:37.468000 | 2011-03-08 11:46:55.875000 | 2011-03-08 12:21:33.343000 |
| 0:43:47.261000 | 2011-03-11 01:34:04.423000 | 2011-03-11 02:17:51.684000 |
| 1:58:49.531000 | 2011-03-21 22:29:59.812000 | 2011-03-22 00:28:49.343000 |
| 0:32:58.172000 | 2011-05-22 14:47:49.265000 | 2011-05-22 15:20:47.437000 |
| 1:24:56.094000 | 2011-05-22 15:22:00.593000 | 2011-05-22 16:46:56.687000 |
| 0:35:08.703000 | 2011-05-23 01:41:37.265000 | 2011-05-23 02:16:45.968000 |
| 1:33:51 | 2011-05-26 15:10:59.171000 | 2011-05-26 16:44:50.171000 |
| 0:35:32.250000 | 2011-05-27 11:12:09.718000 | 2011-05-27 11:47:41.968000 |
| 1:59:32.969000 | 2011-05-27 14:32:58.031000 | 2011-05-27 16:32:31 |
| 0:37:09.484000 | 2011-05-28 13:36:25.609000 | 2011-05-28 14:13:35.093000 |
| 0:32:27.953000 | 2011-05-28 20:38:54.312000 | 2011-05-28 21:11:22.265000 |
| 0:35:36.719000 | 2011-05-30 12:44:12.140000 | 2011-05-30 13:19:48.859000 |
| 0:47:30 | 2011-05-30 13:28:20.609000 | 2011-05-30 14:15:50.609000 |
| 1:46:18.140000 | 2011-05-31 12:21:49.953000 | 2011-05-31 14:08:08.093000 |
| 0:45:14.281000 | 2011-05-31 14:23:42.609000 | 2011-05-31 15:08:56.890000 |

 154

#### 5.2.3.2 User evaluation

**Figure 5.2.4** Screenshot of page 1 of the questionnaire




Questionnaire page 1

Gender: In total, 32 individuals gave their gender: women (17), men (15). There was a good distribution of age.

Age: Participants age ranged from 20 to 60 years old, although more than half where between 30-40 years old. More than a quarter of participants was over 40 years old; see Table 5.5.2, and see Figure 5.5.2.

Games played: Around three fifth of respondents had played none or few RPG or FPS games before; see Table 5.5.3 and see Figure 5.5.3. Only around one fifth had played 10 or more RPG or FPS games before.

 155

**Table 5.2.3** A table of participants' ages

| Age | responses | % |
|---|---|---|
| 0-19 | 0 | 0 |
| 20-29 | 6 | 17.64 |
| 30-39 | 19 | 55.88 |
| 40-49 | 6 | 17.64 |
| 50-59 | 3 | 8.82 |
| 60+ | 0 | 0 |
| Total | 34 | 100 |

**Figure 5.2.5** Barplot of participants' ages

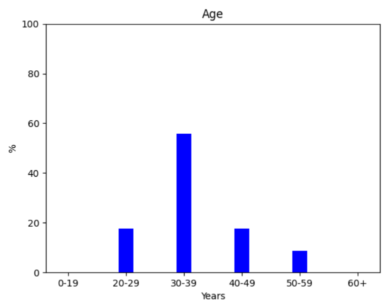
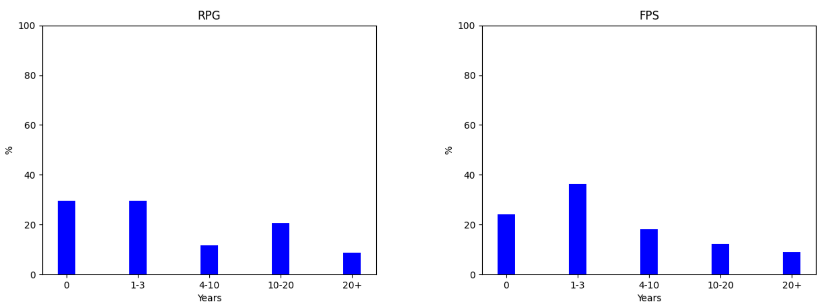



**Table 5.2.4** A table showing the number and percentage of RPGs and FPSs that participants had played.

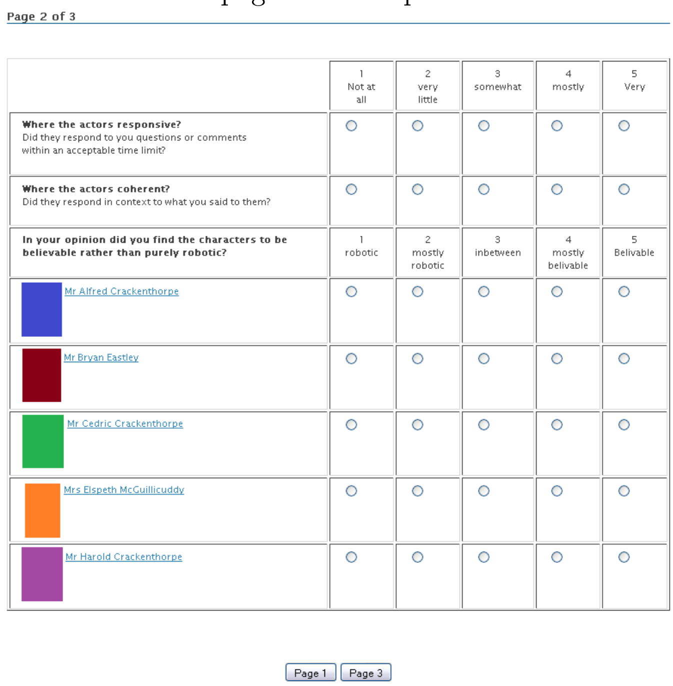
| Option | RPG | % | FPS | % |
|---|---|---|---|---|
| 0 | 10 | 29.41 | 8 | 24.24 |
| 1-3 | 10 | 29.41 | 12 | 36.36 |
| 4-10 | 4 | 11.76 | 6 | 18.18 |
| 10-20 | 7 | 20.58 | 4 | 12.12 |
| 20+ | 3 | 8.82 | 3 | 9.09 |
| Total | 34 | 100 | 33 | 100 |

**Figure 5.2.6** Barplots of the number of RPGs (left) and FPS (right) played by participants



 156

**Figure 5.2.7** Screenshot of page 2 of the questionnaire




Questionnaire page 2

Question: "Where the actors responsive?" *Did they respond to you questions or comments within an acceptable time limit?*
Most respondents, 73.52% found the actors to be mostly or very responsive, see Table 5.5.4, and see Figure 5.5.5.

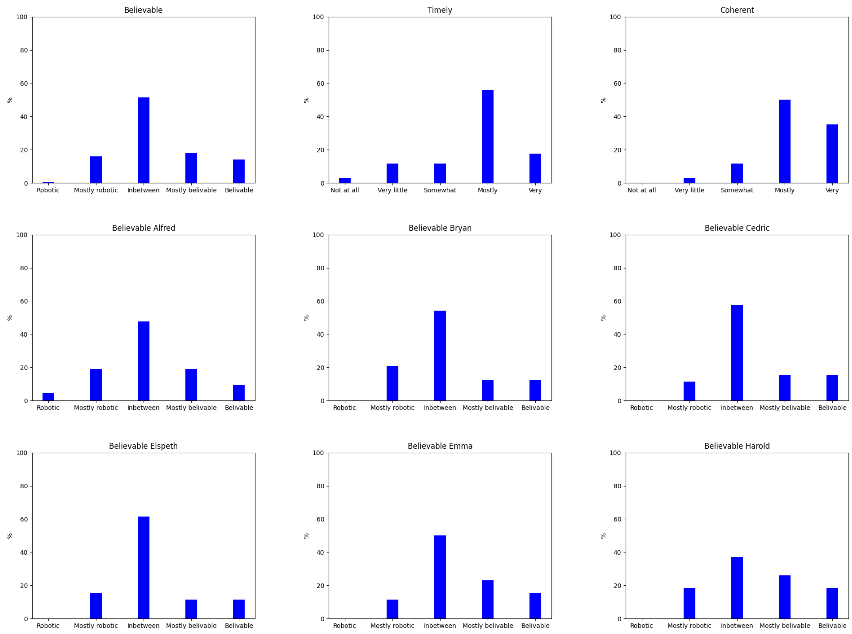
Question: "Where the actors coherent?" *Did they respond in context to what you said to them?*
Most respondents, 85.29%, considered the actors to be mostly or very coherent, see Table 5.5.4, and see Figure 5.5.5.

**Table 5.2.5** A table showing the number and percentage of replies to the actors' timeliness and coherent responses.

| Option | Timely | % | Coherent | % |
|---|---|---|---|---|
| Not at all | 1 | 2.9 | 0 | 0 |
| Very little | 4 | 11.76 | 1 | 2.9 |
| Somewhat | 4 | 11.76 | 4 | 11.76 |
| Mostly | 19 | 55.88 | 17 | 50 |
| Very | 6 | 17.64 | 12 | 35.29 |
| Total | 34 | 100 | 34 | 100 |

 157

**Figure 5.2.8** Barplots that show how believable, timely and coherent the players found the characters overall. Also how believable they found each individual character




In our definition of emergent interactive drama we say that a believable character is both timely and coherent. Additionally in our hypothesis we say that for realtime applicable systems the actors need to be sufficiently responsive to players' actions with a response that is in context to the players' actions according to players' evaluation.

Our null-hypothesis is that players would have found the actors to be not timely or coherent and primarily responded with either "Not at all" or "Very little". Similarily they would have found them to be "robotic" or "mostly robotic" to reflect those responses we generated a Gaussian normal distribution with 0.5 standard deviation for both of these responses; see Figure 5.5.6.

We compared these null-hypothesis against the responses from the players with $p < 0.05$ as significant difference, see Table 5.5.5.

There is a significant difference between the null-hypothesis and the players responses.

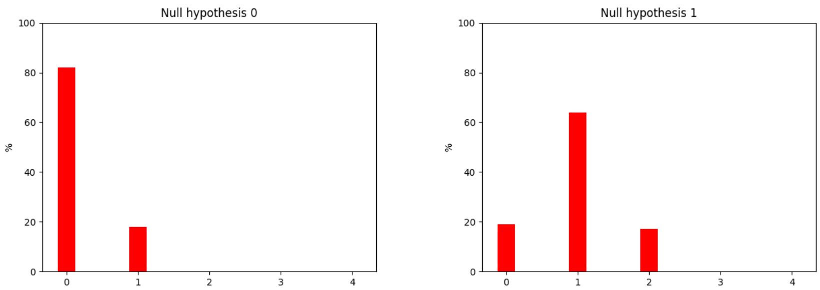
Therefore we reject the null-hypothesis and say that the players found the actors to be both timely and coherent.

**Table 5.2.6** A table showing a significant difference between the null hypothesis and the players responses.

| **Test** | **Null hypothesis** | **t** | **p** | **Significant** |
|---|---|---|---|---|
| Timely | not at all | -21.48464 | 0.00000 | True |
| Timely | very little | -12.26684 | 0.00000 | True |
| Coherent | not at all | -29.87181 | 0.00000 | True |
| Coherent | very little | -17.15571 | 0.00000 | True |
| Believable | robotic | -18.13663 | 0.00000 | True |
| Believable | mostly robotic | -9.53956 | 0.00000 | True |
| Believable Alfred | robotic | -14.80357 | 0.00000 | True |
| Believable Alfred | mostly robotic | -6.78671 | 0.00000 | True |
| Believable Bryan | robotic | -16.53351 | 0.00000 | True |
| Believable Bryan | mostly robotic | -7.75474 | 0.00000 | True |
| Believable Cedric | robotic | -18.61588 | 0.00000 | True |
| Believable Cedric | mostly robotic | -9.24743 | 0.00000 | True |
| Believable Elspeth | robotic | -17.77378 | 0.00000 | True |
| Believable Elspeth | mostly robotic | -8.34516 | 0.00000 | True |
| Believable Emma | robotic | -19.14982 | 0.00000 | True |
| Believable Emma | mostly robotic | -9.72801 | 0.00000 | True |
| Believable Harold | robotic | -18.13663 | 0.00000 | True |
| Believable Harold | mostly robotic | -9.53956 | 0.00000 | True |

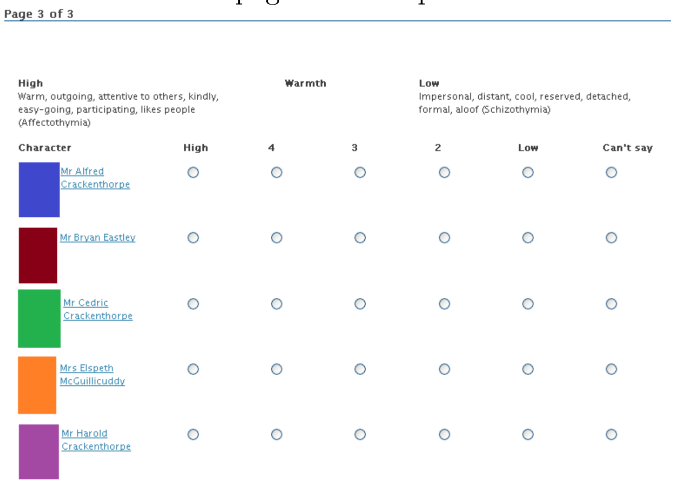
 159

**Figure 5.2.9** Barplots that show null hypothesis “Not at all”/“Robotic”(left) and “Very little”/“Mostly robotic”(right) players found the actors



 160

**Figure 5.2.10** Screenshot of page 3 of the questionnaire




Questionnaire page 3 On page 3 we asked about the 7 Cattell's traits that we had used. The screenshot in Figure 5.5.7 shows how we asked about the trait; naming both the low and high end of the trait with a brief description and the player was asked to indicate for each character.

**Table 5.2.7** A table showing convergence towards character traits.

| **Character** | **Trait** | **t** | **p** |
|---|---|---|---|
| Alfred | Reserved | -2.51738 | 0.01312 |
| Alfred | Serious | -1.95845 | 0.05246 |
| Alfred | Rule-conscious | 2.58678 | 0.01092 |
| Bryan | Dominant | 1.01679 | 0.31120 |
| Cedric | Lively | 3.15795 | 0.00197 |
| Emma | Sensitive | 1.96462 | 0.05170 |
| Harold | Emotionally stable | 3.17356 | 0.00191 |
| Harold | Dominant | 2.06205 | 0.04126 |
| Harold | Serious | -3.07975 | 0.00254 |
| Harold | Rule-conscious | 2.87256 | 0.00482 |

## 5.3 Mistlethorpe

Mistlethorpe is a web-based game that we presented at Venturefest in York, England. It is a web-based game with the first full mystery implemented. The player enters an English village called Mistlethorpe and receives an invitation to the village mansion where all is not as tranquil as it first appears. The player gets access to the whole mansion and can go between rooms and interrogate the suspect.

Soon after the player enters the mansion and talks to the hosts, someone finds the cook murdered on the kitchen floor. The player can now investigate the murder scene and travel around the mansion.

We added narrative messages, for example, when the player approaches the mansion, the engine offers the player two options, see Figure 5.3.1: Whether the mansion is:

i. mansion is beautiful or
ii. ostentatious?

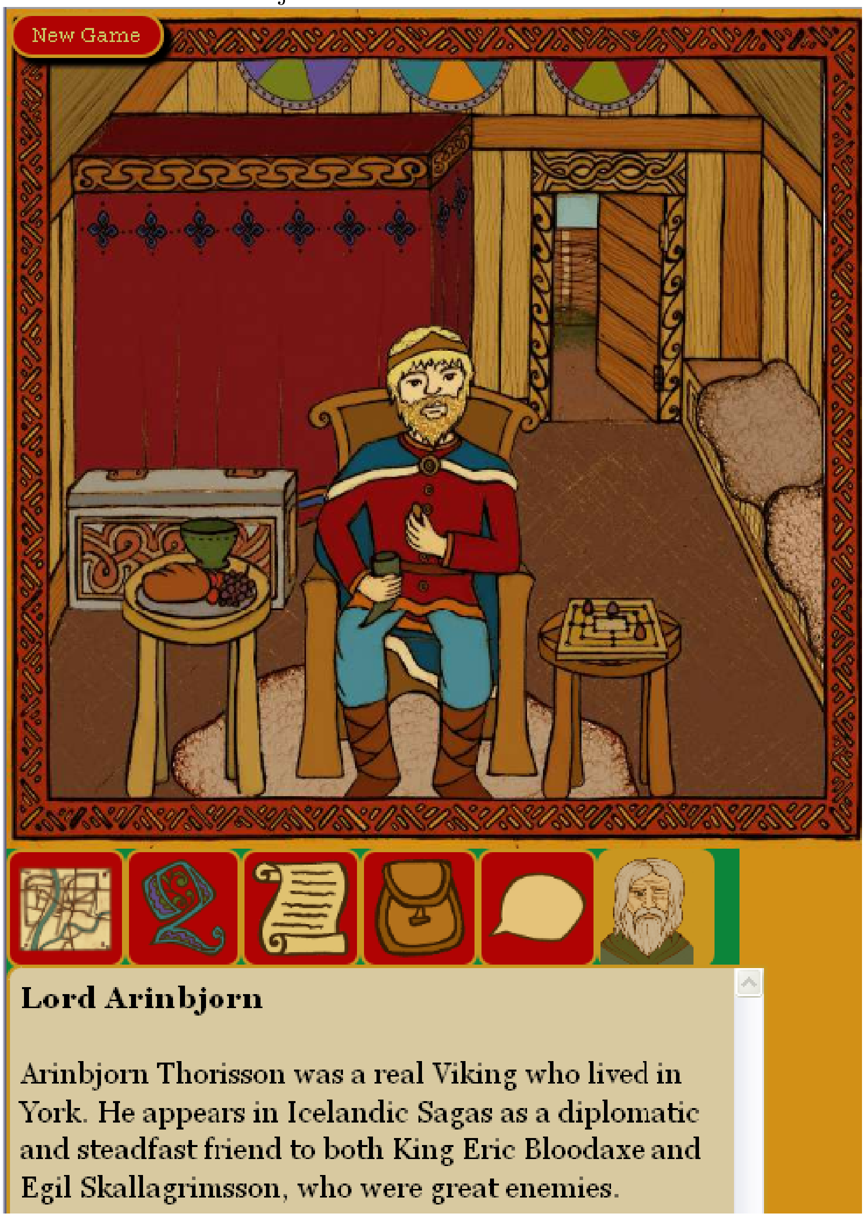
**Figure 5.3.1** The Mistlethorpe mansion


Several events take place to increase the suspense and finally, over supper, the detective discovers the murderer.

## 5.4 Merchants of Jorvik

Merchants of Jorvik is a game we created for the Jorvik Viking Museum. It is a web-based game optimised for mobile devices—specifically iPhones. According to Egils saga, Skallagrímssonar, then Arinbjörn was a lord in York during the Viking era. The game starts in Lord Arinbjorn's house, see Figure 5.5.1

**Figure 5.4.1** Lord Arinbjorn




In this game, we developed new roles, actions and events. The game offers two quests:

i. A haggling quest; the player is given a set of objects to buy and must haggle down the prices. For this, the player must travel around Jorvik, see Figure 5.4.2 and haggle with merchants on their stalls, see Figure see Figure 5.4.3.
ii. Find a thief that stole a valuable brooch.

The interaction between player and actor agents is in dialogue and physical actions:

- The player is presented with a dynamic list of multiple choices of relevant sentences to choose from.
- The player can pick up and examine any object made available in the scene.
- The player can travel between scenes.
- The player can initiate conversation verbally.
- The player can initiate conversation via physical actions such as examining an object.
- The actors can initiate conversation with the player.

In Mistelthorp, see Section 5.3, and Merchants of Jorvik, see Section 5.4, web games we offer key words rather than sentences to the user which reduces the feeling of clutter, see example from the Merchants of Jorvik, see Figure 5.4.4

We show the results in Figure 5.4.5, the player is in the study and is interrogating Cedric. Bryan and Elspeth are also present, but neither Alfred nor Emma is present. The player can also talk about Emma and Alfred with the actors that are in the room. Selecting the option 'Talk about' will offer similar sentences, but instead of asking about Cedric, it asks Cedric about Alfred in this case. For example "Are you rich?" becomes "Is Alfred rich?".

All the sentences are generated based on the BN, by looking at parent and children vertices and leaving the decision of what to say to the player.

 164

**Figure 5.4.2** Map of Jorvik

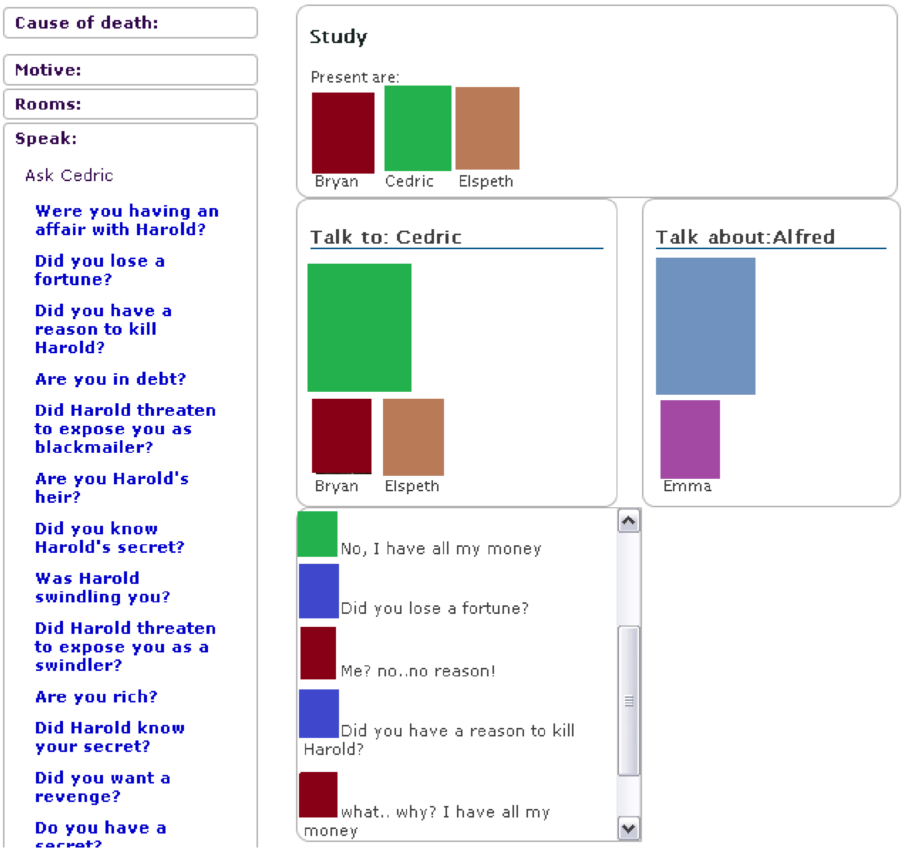
 165

**Figure 5.4.3** The weaver's stall

**Figure 5.4.4** Key words as options rather than whole sentences

167

**Figure 5.4.5** A range of speech acts for the player



## 5.5 User evaluation

#### 5.5.0.1 Computational evaluation

**Playtime,** A total of 34 participants answered the questionnaire, from the logs we can gain some information in addition to what they answer. We can filter the logs for those that played for more than half an hour and for more than an hour. Some play for a while and then stopped for an hour and came back, and it is dicfficult to say how log those plaeyd in total. So as not to inflate the c=values we filtered out anyone who had spent more than 2 hours as those ar mostly people that didn't play for that entire time.

Therefore we filterd out those that had played between 30 minuted - 2 hours.

i. > 30 minute: 28 of 34
ii. > 1 hour: 9 of 34
iii. > 1 hour and 30 minutes: 7 of 34

Further more the 4 years of logs that we have accumulated and give access to in the code repository demonstrate that the characters answer within a second.

#### 5.5.0.2 User evaluation

**Questionnaire page 1**

**Gender:** In total, 32 individuals gave their gender: women (17), men (15). There was a good distribution of age.

**Age:** Participants age ranged from 20 to 60 years old, although more than half where between 30-40 years old. More than a quarter of participants was over 40 years old; see Table 5.5.2, and see Figure 5.5.2.

**Games played:** Around three fifth of respondents had played none or few RPG or FPS games before; see Table 5.5.3 and see Figure 5.5.3. Only around one fifth had played 10 or more RPG or FPS games before.

**Table 5.5.1** Participants that played the game between half an hour to two hours

| **Playtime** | **Start time** | **end time** |
|---|---|---|
| 0:49:45.203000 | 2010-10-18 13:22:20 | 2010-10-18 14:12:05.203000 |
| 0:31:25.672000 | 2010-10-18 22:52:44.921000 | 2010-10-18 23:24:10.593000 |
| 0:32:14.484000 | 2010-10-19 22:25:36.687000 | 2010-10-19 22:57:51.171000 |
| 0:32:58.110000 | 2010-10-20 12:34:06.921000 | 2010-10-20 13:07:05.031000 |
| 0:34:17.515000 | 2010-11-03 12:34:51.750000 | 2010-11-03 13:09:09.265000 |
| 1:16:50.594000 | 2010-11-03 23:38:54.359000 | 2010-11-04 00:55:44.953000 |
| 0:30:39.359000 | 2010-11-05 01:40:24.578000 | 2010-11-05 02:11:03.937000 |
| 1:44:54.485000 | 2010-11-05 11:36:57.593000 | 2010-11-05 13:21:52.078000 |
| 1:14:07.625000 | 2010-12-14 13:33:33.093000 | 2010-12-14 14:47:40.718000 |
| 0:51:21.047000 | 2010-12-16 11:38:57.843000 | 2010-12-16 12:30:18.890000 |
| 0:36:53.907000 | 2011-01-24 13:38:05.921000 | 2011-01-24 14:14:59.828000 |
| 0:41:44.672000 | 2011-02-02 23:23:18.164000 | 2011-02-03 00:05:02.836000 |
| 1:52:41.313000 | 2011-02-08 10:55:39.515000 | 2011-02-08 12:48:20.828000 |
| 0:34:37.468000 | 2011-03-08 11:46:55.875000 | 2011-03-08 12:21:33.343000 |
| 0:43:47.261000 | 2011-03-11 01:34:04.423000 | 2011-03-11 02:17:51.684000 |
| 1:58:49.531000 | 2011-03-21 22:29:59.812000 | 2011-03-22 00:28:49.343000 |
| 0:32:58.172000 | 2011-05-22 14:47:49.265000 | 2011-05-22 15:20:47.437000 |
| 1:24:56.094000 | 2011-05-22 15:22:00.593000 | 2011-05-22 16:46:56.687000 |
| 0:35:08.703000 | 2011-05-23 01:41:37.265000 | 2011-05-23 02:16:45.968000 |
| 1:33:51 | 2011-05-26 15:10:59.171000 | 2011-05-26 16:44:50.171000 |
| 0:35:32.250000 | 2011-05-27 11:12:09.718000 | 2011-05-27 11:47:41.968000 |
| 1:59:32.969000 | 2011-05-27 14:32:58.031000 | 2011-05-27 16:32:31 |
| 0:37:09.484000 | 2011-05-28 13:36:25.609000 | 2011-05-28 14:13:35.093000 |
| 0:32:27.953000 | 2011-05-28 20:38:54.312000 | 2011-05-28 21:11:22.265000 |
| 0:35:36.719000 | 2011-05-30 12:44:12.140000 | 2011-05-30 13:19:48.859000 |
| 0:47:30 | 2011-05-30 13:28:20.609000 | 2011-05-30 14:15:50.609000 |
| 1:46:18.140000 | 2011-05-31 12:21:49.953000 | 2011-05-31 14:08:08.093000 |
| 0:45:14.281000 | 2011-05-31 14:23:42.609000 | 2011-05-31 15:08:56.890000 |

**Table 5.5.2** A table of participants' ages

| Age | responses | % |
|---|---|---|
| 0-19 | 0 | 0 |
| 20-29 | 6 | 17.64 |
| 30-39 | 19 | 55.88 |
| 40-49 | 6 | 17.64 |
| 50-59 | 3 | 8.82 |
| 60+ | 0 | 0 |
| Total | 34 | 100 |

 170

**Figure 5.5.1** Screenshot of page 1 of the questionnaire

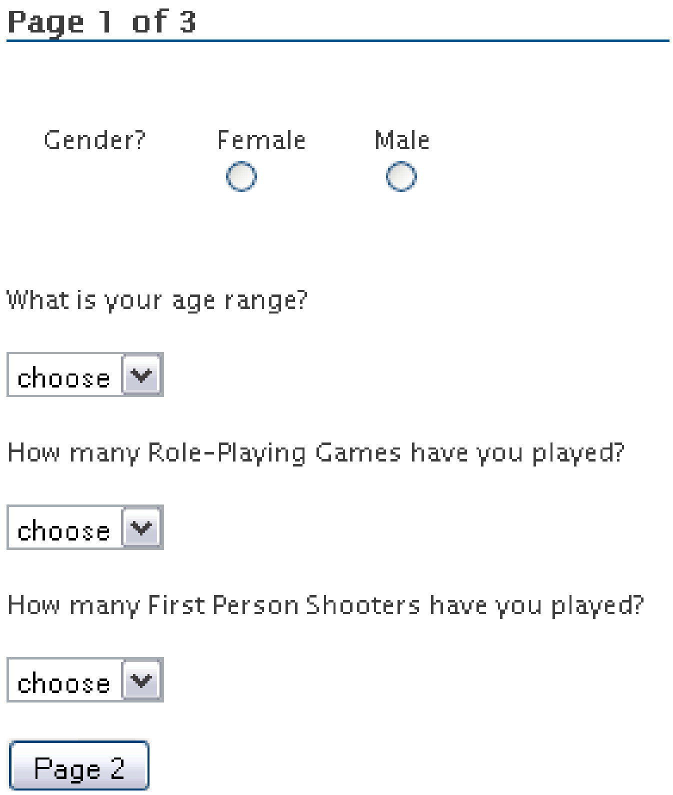



**Figure 5.5.2** Barplot of participants' ages

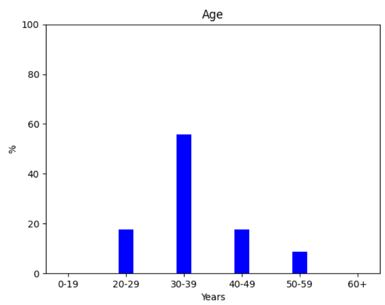
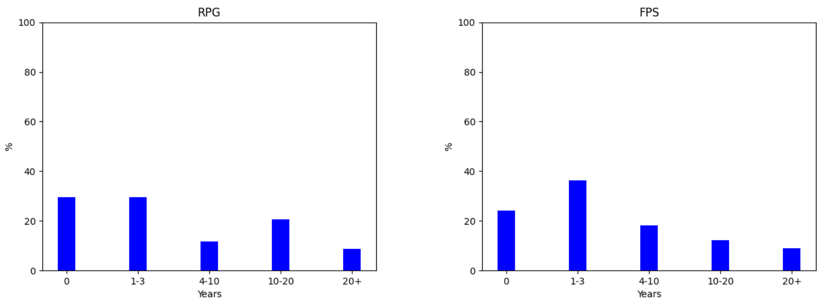



**Table 5.5.3** A table showing the number and percentage of RPGs and FPSs that participants had played.

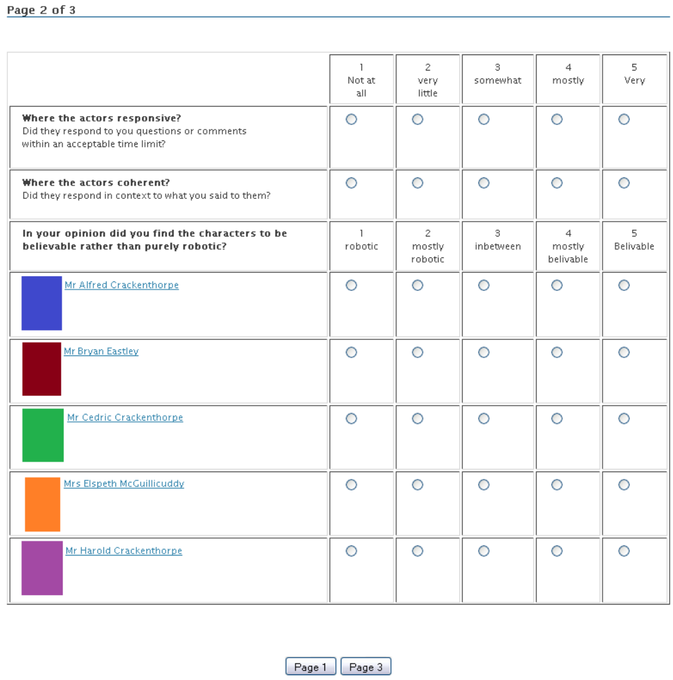
| Option | RPG | % | FPS | % |
|---|---|---|---|---|
| 0 | 10 | 29.41 | 8 | 24.24 |
| 1-3 | 10 | 29.41 | 12 | 36.36 |
| 4-10 | 4 | 11.76 | 6 | 18.18 |
| 10-20 | 7 | 20.58 | 4 | 12.12 |
| 20+ | 3 | 8.82 | 3 | 9.09 |
| Total | 34 | 100 | 33 | 100 |

**Figure 5.5.3** Barplots of the number of RPGs (left) and FPS (right) played by participants



 172

**Figure 5.5.4** Screenshot of page 2 of the questionnaire




Questionnaire page 2

Question: "Where the actors responsive?" *Did they respond to you questions or comments within an acceptable time limit?*
Most respondents, 73.52% found the actors to be mostly or very responsive, see Table 5.5.4, and see Figure 5.5.5.

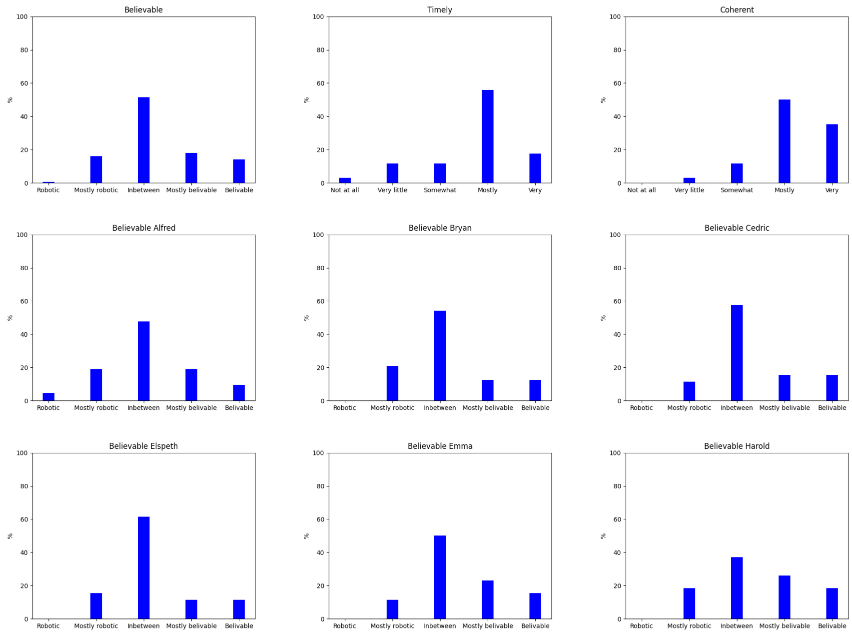
Question: "Where the actors coherent?" *Did they respond in context to what you said to them?*
Most respondents, 85.29%, considered the actors to be mostly or very coherent, see Table 5.5.4, and see Figure 5.5.5.

**Table 5.5.4** A table showing the number and percentage of replies to the actors' timeliness and coherent responses.

| Option | Timely | % | Coherent | % |
|---|---|---|---|---|
| Not at all | 1 | 2.9 | 0 | 0 |
| Very little | 4 | 11.76 | 1 | 2.9 |
| Somewhat | 4 | 11.76 | 4 | 11.76 |
| Mostly | 19 | 55.88 | 17 | 50 |
| Very | 6 | 17.64 | 12 | 35.29 |
| Total | 34 | 100 | 34 | 100 |

 173

**Figure 5.5.5** Barplots that show how believable, timely and coherent the players found the characters overall. Also how believable they found each individual character




In our definition of emergent interactive drama we say that a believable character is both timely and coherent. Additionally in our hypothesis we say that for realtime applicable systems the actors need to be sufficiently responsive to players' actions with a response that is in context to the players' actions according to players' evaluation.

Our null-hypothesis is that players would have found the actors to be not timely or coherent and primarily responded with either "Not at all" or "Very little". Similarily they would have found them to be "robotic" or "mostly robotic" to reflect those responses we generated a Gaussian normal distribution with 0.5 standard deviation for both of these responses; see Figure 5.5.6.

We compared these null-hypothesis against the responses from the players with $p < 0.05$ as significant difference, see Table 5.5.5.

There is a significant difference between the null-hypothesis and the players responses.

Therefore we reject the null-hypothesis and say that the players found the actors to be both timely and coherent.

**Table 5.5.5** A table showing a significant difference between the null hypothesis and the players responses.

| Test | Null hypothesis | t | p | Significant |
|---|---|---|---|---|
| Timely | not at all | -21.48464 | 0.00000 | True |
| Timely | very little | -12.26684 | 0.00000 | True |
| Coherent | not at all | -29.87181 | 0.00000 | True |
| Coherent | very little | -17.15571 | 0.00000 | True |
| Believable | robotic | -18.13663 | 0.00000 | True |
| Believable | mostly robotic | -9.53956 | 0.00000 | True |
| Believable Alfred | robotic | -14.80357 | 0.00000 | True |
| Believable Alfred | mostly robotic | -6.78671 | 0.00000 | True |
| Believable Bryan | robotic | -16.53351 | 0.00000 | True |
| Believable Bryan | mostly robotic | -7.75474 | 0.00000 | True |
| Believable Cedric | robotic | -18.61588 | 0.00000 | True |
| Believable Cedric | mostly robotic | -9.24743 | 0.00000 | True |
| Believable Elspeth | robotic | -17.77378 | 0.00000 | True |
| Believable Elspeth | mostly robotic | -8.34516 | 0.00000 | True |
| Believable Emma | robotic | -19.14982 | 0.00000 | True |
| Believable Emma | mostly robotic | -9.72801 | 0.00000 | True |
| Believable Harold | robotic | -18.13663 | 0.00000 | True |
| Believable Harold | mostly robotic | -9.53956 | 0.00000 | True |

 175

**Figure 5.5.6** Barplots that show null hypothesis “Not at all”/“Robotic”(left) and “Very little”/“Mostly robotic”(right) players found the actors

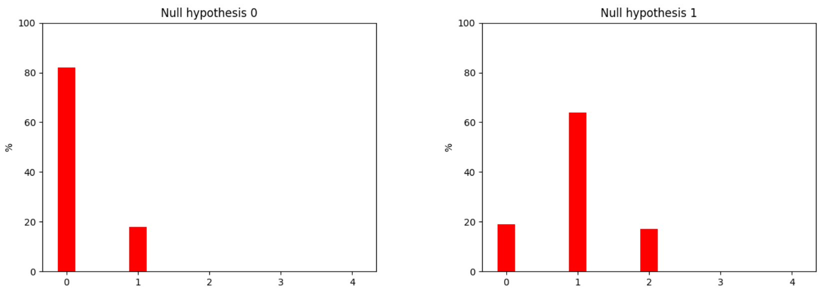
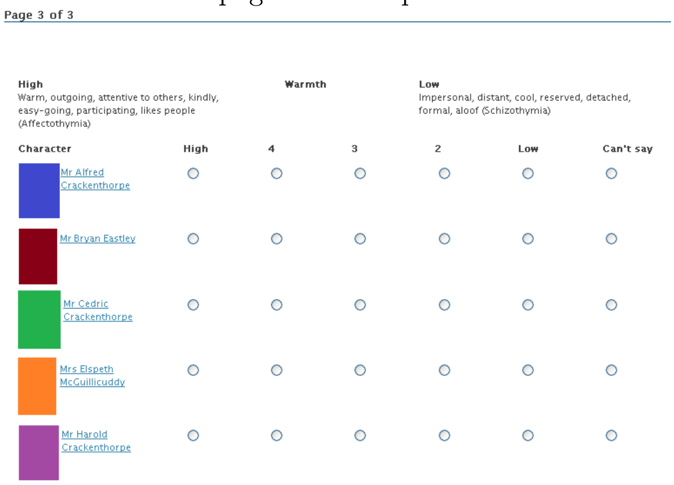


**Figure 5.5.7** Screenshot of page 3 of the questionnaire




Questionnaire page 3 On page 3 we asked about the 7 Cattell's traits that we had used. The screenshot in Figure 5.5.7 shows how we asked about the trait; naming both the low and high end of the trait with a brief description and the player was asked to indicate for each character.

**Table 5.5.6** A table showing convergence towards character traits.

| **Character** | **Trait** | **t** | **p** |
|---|---|---|---|
| Alfred | Reserved | -2.51738 | 0.01312 |
| Alfred | Serious | -1.95845 | 0.05246 |
| Alfred | Rule-conscious | 2.58678 | 0.01092 |
| Bryan | Dominant | 1.01679 | 0.31120 |
| Cedric | Lively | 3.15795 | 0.00197 |
| Emma | Sensitive | 1.96462 | 0.05170 |
| Harold | Emotionally stable | 3.17356 | 0.00191 |
| Harold | Dominant | 2.06205 | 0.04126 |
| Harold | Serious | -3.07975 | 0.00254 |
| Harold | Rule-conscious | 2.87256 | 0.00482 |

## 5.6 Computational drama evaluation

### 5.6.1 Drama interestingness

N. Szilas & Co proposes a computational model to measure interestingness in drama [SER16], which is very promising and useful work since it is challenging to measure complex algorithms with user evaluation accurately; there is minimal availability of comparison results. Scientists are not necessarily, even rarely, also game developers or proficient artists. It is, therefore, very constructive to develop computational models to measure our systems.

We looked at the Afghan poppy scenario with the modelling suggested by N. Szilas & Co. and stepped through a scenario as follows:

i. We say that a positive conflict, +1, is generated when an actor prioritises a character goal over a drama goal.
ii. We also say that a deceiving character, for example, the farmer deceiving the officer in Section 4.5.4.2 is also a positive increase, +1.
iii. Furthermore, when the actor offers another actor the drama goal, and the other actor accepts, it is also positive, +1.
iv. A new character has entered the scene, which is also a +1 in interestingness.
v. Furthermore, the new character is likely related to the farmer, another +1 in interestingness.
vi. The new character also has their own character goals, which add still another +1.

At this point, we have already 6 points as a result of one character deception, and we can go on like this, which has caused us some pause.

When we start to run the DED against large data sources, the drama could become to complex for the player effectively grasp.

## 5.7 Summary

# 6
# Conclusion

**Real-time applicable** For all the applications, response time was well within a second, as discussed in the evaluation we have 4 years of logs that demonstrate this. We were also able to have the actors respond within a second in the Second Life despite considerable overhead caused by other resources required to run the actors in the Second Life virtual world. In the latest experiment, we ran it using the SpatialOS technology stack, and all actor decisions take less than a second.

We intend to test the performance further using much larger BNs than we have up to now. The current plan is to use automated processes to create BN from real-world data to use in synthetic training.

**that include several characters, behaviours and scenes and run for extended periods** we have implemented five distinctly different scenarios all with different characters and scenes without any significant overhead. Characters are defined abstractly as roles that have certain base traits; for example, an officer is likely conscientious. The characters do not need to have specific instructions or scripts to interact with each other, the goals are not specific to characters, instead, they are specific to roles, and we can, therefore, try a similar scenario with multiple different characters in the designated roles to create varied interactions and conflicts between different personality types.

We gave evidence in the evaluation that the drama plays for an extended period of time, some playing for over an hour.

## 6.1 Limitations

We are aware that our tests are on small manually configure data sets and that when we start to test the system on more extensive data sets, primarily auto-sourced sets. We may encounter unexpected side effects of decentralised drama management. When looking at the system with computational modelling in Section **??**, we saw an advanced warning of this possibility, which is something we will be looking at further.

We also know that the system is not suitable for scenarios where a specific storyline is required since there is minimal top-down author control. We are not aiming for applications that require pre-authored stories. We aim to test the DED further in a synthetic training environment where it is more important that the system behaves realistically and imitates opponents behaviour based on real-world data.

## 6.2 Future work

We find our tests to date very promising and are looking to try more advanced examinations on a more extensive database with real-world data and natural language processing. We will also look at computational modelling based on N. Szilas & Co to evaluate interestingness and other factors. For example, to understand the situations formed during training scenarios and when designing new scenes. To see if we can use such modelling to evaluate the scenario's training value to name a few possible avenues for future testing of the DED. We are currently working with Improbable Defence department on future emphasis.

# Glossary

**A**

**actor** Autonomous agents with deep Artificial Intelligence (AI) that play characters in emergent drama. 112–114, 118–120, 122, 123, 128–130, 136, 139, 149

**autonomous agent** A primarily self governed AI agent; acts without direct human control. 3, 159

**B**

**Bayesian Network** Belief Networks structured as a directed acyclic graph (DAG) that use Bayes rule to evaluate its probabilities. iv, 9, 25, 50, 160, 163

**Belief Network** Probabilistic graphs comprised of vertices and directed arcs that indicate conditional dependency. 25

**believability** The degree to which an autonomous agent simulates human behaviour. 3, 25

**C**

**character** A character role in a drama. 113, 115–118, 120, 122, 130, 133

**character goal** Goals for characters in emergent drama that are specific for the character role to stay in-character. 67, 113, 116, 139

**coherent response** respond in a logical context to past and present events, their role, character, and emotional state. 3

**Compellingness criteria** A criteria to measure the success of an interactive drama system [Bar08]. 7

**computer game** A game played on a computer, including: personal computers, consoles, and mobile devices. 161

**Computer-Based Role-Playing Game** An RPG as a computer game. 21, 163

**D**

 181

**D-separation** The blocking of the flow of evidence between vertices, see Definition 2.3.3. iv–vi, 32, 35–38, 50, 135, 136, 142

**directed acyclic graph** A graph that has directed arcs and no cycles. 25, 159, 163

**director** Autonomous agent, a drama manager that guides emergent drama with schemas. 119

**drama** A play for theatre, radio, television, and computer games that has a set of characters, and a plot. i, 5–7, 13–17, 19, 20, 23, 25, 52, 112, 116, 122, 130, 139, 146, 161

**drama goal** Goals that structure emergent drama and guide the actors in developing the plot. 67, 98, 113, 149

**drama proper** A drama for a theatre stage as described in the Oxford English Dictionary:

> *" A composition in prose or verse, adapted to be acted on stage, in which a story is related by means of dialogue and action, and is represented, with accompanying gesture, costume and scenery, as in real life; a play. "*
>
> – [Dic12]

. 14–16

**Dynamic Plot Generating Engine** An engine that creates new plots dynamically. 9, 51, 163

## E

**emergent interactive drama** See Definition 1.1.1. vi, 2–4, 6, 7, 24, 68, 112

**equilibrium** See Definition 2.3.7. 43

**expected utility** The calculated value of an action or state given observed evidence, see Definition 2.3.5. vi, 39, 161, 163

## G

**Game Theory** Game Theory. 9, 25, 50, 163

**GeNIe** A user interface to design and display Bayesian Networks (BNs), [DSL07a]. iv, 163

## I

**in character** An actor or player that is playing a character in a play or a game and act as that character. 17, 164

**interactive drama** In this thesis, interactive drama refers to drama in computer games, virtual realitys (VRs), and simulations, where the player plays a critical role that affects the course of the drama. i, 2, 5, 19, 20, 69, 70, 76–78, 81, 82, 84, 96, 102, 146, 149

**M**

**Massively Multiplayer Online Game** A computer game played online by millions of online players. v, 162, 164

**Multi-Agent Influence Diagram** Multi-Agent Influence Diagram. vii, 164

**N**

**narrative** A structured sequence of real or unreal events. 9

**Nash Equilibrium** A method of calculating an optimal strategy that has an expected utility (EU) that is equal or higher to any other EU in the game. 3

**Non-Player Character** Characters in computer games that are played by the computer. i, 164

**O**

**Object-Oriented Bayesian Networks** A method of dividing the network up into manageable pieces in order to construct a much lager network through the reuse of the individual objects. v, 98, 164

**officer** A military personel with command over other personel, in the context of the the Afghan poppy scenario, an officer is second in command to the OC. 138

**Officer Commanding** An officer in command of a minor unit in the British Army. 99, 164

**out of character** An actor or player that is playing a character in a play and starts to behave in a manner that is uncharacteristic for the character they are playing. 112, 164

**P**

**Pen & Paper Role-Playing Game** An RPG played in a group with pen & paper and dice to decide game events. 21, 164

**player** A person playing a game. i, 1–3, 52, 70, 112, 116, 119, 130, 149, 161

**R**

**radio drama** A purely acoustic dramatisation for radio broadcast. 15

**rational** rational. 3

**Rational Dialogue Engine** An engine that creates rational dialogues. 6, 9, 164

**rationality** Rationality is choosing the strategy that will give maximum EU no matter what the opponents play, given current information, see Definition 2.3.8.. vi, 44

**Role-Playing Game** A game where the players take on a role in a fantasy adventure. 21, 164

**S**

**Second Life [Lab03]** A VR that allows developers to programm agents that can joint the world and interact with the game world as the players plural. 1, 164

**SpatialOS** A cloud service tool that manages the complexity and scaling difficulties of distributing Massively Multiplayer Online Game (MMOG) over multiple servers and maintaining a single vast game world. 8, 140, 162

**T**

**the Afghan poppy scenario** A purely academic scenario, based loosely on historic concerns, constructed to demonstrate the Directed Emergent Drama (DED) and test it in with SpatialOS. 161

**V**

**vertex** Vertex in a graph. iii–vii, 25–33, 36–41, 49, 53, 56–61, 63, 66, 105, 118–125, 128–131, 133, 135, 136, 138, 142, 159

**virtual reality** According to Wikipedia:

> *" A virtual reality is a simulated experience that can be similar to or completely different from the real world. "*
>
> – [Wik21]

. 1, 161, 164

# Acronyms

**A**

**AI** Artificial Intelligence. 159

**B**

**BN** Bayesian Network. iv, v, 6, 9, 25, 27, 28, 30–33, 37–40, 49–57, 66, 78, 81, 96, 97, 105, 110–113, 119, 120, 124, 129, 133, 138, 142–148, 160

**BSc** Bachelor of Science. 9

**C**

**CB-RPG** Computer-Based Role-Playing Game. 21, 22, 66

**D**

**DAG** directed acyclic graph. 25, 71, 72, 159

**DED** Directed Emergent Drama. v, 5, 8, 9, 25, 50, 51, 66, 96, 97, 102, 105, 110, 112, 140, 143, 144, 146–149, 162

**DPGE** Dynamic Plot Generating Engine. 9, 51–57, 66, 98, 104, 109, 119, 147, 148

**E**

**EU** expected utility. vi, vii, 39, 43, 123–128, 131–135, 148, 161

**G**

**GeNIe** GeNIe. iv–vi, 27, 28, 30, 33–36, 38, 40, 41, 54, 58, 60, 61, 97, 110, 111, 121, 124–126, 128, 131, 132, 136, 137

**GT** Game Theory. 9, 25, 42, 44, 45, 50, 65, 66, 70, 97, 108, 112, 145, 148

**I**

**IC** in character. 17, 98, 144, 149

**M**

**MAID** Multi-Agent Influence Diagram. vii, 25, 49–51, 66, 97, 99, 112, 113, 120, 144

 185

**MMOG** Massively Multiplayer Online Game. v, 1, 8, 107, 108, 112, 119, 162

**MSc** Master of Science. 9

**N**

**NPC** Non-Player Character. i, vi, 2, 3, 20–24, 55, 66, 68–70, 73, 93, 97, 119, 140, 141, 144

**O**

**OC** Officer Commanding. 99, 101, 103–105, 110, 114, 119, 120, 122, 129, 138, 140, 143

**OOBN** Object-Oriented Bayesian Networks. v, 38, 50, 67, 98, 110, 111

**OOC** out of character. 112

**P**

**PP-RPG** Pen & Paper Role-Playing Game. 21, 66

**R**

**RDE** Rational Dialogue Engine. 6, 9, 51, 56, 61, 66, 118

**RPG** Role-Playing Game. 21–23, 76, 93, 145, 146

**S**

**SL** Second Life [Lab03]. 1, 151

**V**

**VR** virtual reality. 1, 2, 11, 69, 151, 161, 162

# References


[Aar11] A. Aarne. Verzeichnis der märchentypen. *Folklore Fellows Communications No. 3*, 1911.

[Abb02] H. P. Abbott. *The Cambridge Introduction to Narrative*. Cambridge Introductions to Literature. Cambridge University Press, 1 edition, 2002.

[ABK09] F. Arinbjarnar, H. Barber, and D. Kudenko. A critical review of interactive drama systems. In *AISB'09 Symposium: AI & Games*, Edinburgh, UK, 2009.

[AFL+06a] R. Aylett, R. Figueiredo, S. Louchart, J. Dias, and A. Paiva. Making it up as you go along - improvising stories for pedagogical purposes. In *IVA*, pages 304–315. Springer, 2006.

[AFL+06b] R. Aylett, R. Figuieredo, S. Louchart, J. Dias, and A. Paiva. Towards a narrative theory of virtual reality. In J. Gratch, M. Young, R. Aylett, D. Ballin, and P. Olivier, editors, *Virtual Reality Journal*, volume 7, pages 307–315. Springer, 2006. LNAI 4133.

[AGZ07] Y. Asnar, P. Giorgini, and N. Zannone. Reasoning about risk in agents deliberation process: a jadex implementation. In *In Proceeding of 8th International Workshop on Agent Oriented Software Engineering (AOSE 2007*, 2007.

[AK08] F. Arinbjarnar and D. Kudenko. Schemas in directed emergent drama. In *proceedings of the 1$^{st}$ Joint International Conference on Interactive Digital Storytelling ICIDS08*, Erfurt, Germany, 2008.

[AK09a] F. Arinbjarnar and D. Kudenko. Directed emergent drama vs. pen & paper role-playing games. In *AISB'09 Symposium: AI & Games*, Edinburgh, UK, 2009.

[AK09b] F. Arinbjarnar and D. Kudenko. Duality of actor and character goals in virtual drama. In *9$^{th}$ Interna-*

*tional Conference on Intelligent Virtual Agents*, Amsterdam, Holland, 2009.

[AK10] F. Arinbjarnar and D. Kudenko. Bayesian networks: Real-time applicable decision mechanisms for intelligent agents in interactive drama. In *proceedings of the IEEE Conference on Computational Intelligence and Games (CIG2010)*, Copenhagen, Denmark, 2010.

[AK12] F. Arinbjarnar and D. Kudenko. Actor bots. In P. Hingston, editor, *Believable Bots*, pages 69–97. Springer, 2012.

[AL08] R. Aylett and S. Louchart. If i were you: Double appraisal in affective agents. In *Proceedings of the Autonomous Agents and Multi-agent Systems AAMAS*, 2008.

[ALI06] ALICE. Homepage of the A.L.I.C.E. artificial intelligence foundation. online, Last accessed: 07.31.2006. http://www.alicebot.org (Last accessed: 19.01.2013).

[ALT+08] R. Aylett, S. Louchart, A. Tychsen, M. Hitchens, R. Figueiredo, and C. D. Mata. Managing emergent character-based narrative. In *The Second International Conference on Intelligent Technologies for Interactive Entertainment*, Cancun, Mexico, January 2008.

[ALW11] R. Aylett, S. Louchart, and A. Weallans. Research in interactive drama environments, role-play and storytelling. In *International Conference on Interactive Digital Storytelling ICIDS11*, pages 132–143, 2011.

[Ari06] F. Arinbjarnar. Murder he programmed: A dynamic plot generating engine for murder mystery games. Bachelor of Science project at Reykjavik University, 2006.

[Ari07a] F. Arinbjarnar. Rational dialog in interactive games. Master's thesis, Reykjavik University, 2007.

[Ari07b] F. Arinbjarnar. Rational dialog in interactive games. In *proceedings of AAAI Fall Symposium on Intelligent Narrative Technologies*, Westin Arlington Gateway, Arlington, Virginia, 2007.

[Ari08] F. Arinbjarnar. Dynamic plot generation engine. In *proceedings of the Workshop on Integrating Technologies for Interactive Stories*, Playa del Carmen, Mexico, 2008.

[Art19] Electronic Arts. The sims, 2000-2019.

[Bar08] H. Barber. *Generator of Adaptive Dilemma-based Interactive Narratives (GADIN).* PhD thesis, Department of Computer Science, The University of York, York, England, 2008.

[BB00] G. Ball and J. Breese. Relating personality and behavior: Posture and gestures. In A. Paiva, editor, *Affective Interactions*, pages 196–203. Springer-Verlag, Berlin, 2000.

[BDDF05] R. Bordini, M. Dastani, J. Dix, and A. Seghrouchni E. Fallah. *Multi-Agent Programming: Languages, Platforms and Applications.* Springer-Verlag, Kluwer, Dordrecht, 2005.

[Bec53] S. Beckett. *Waiting for Godot.* Grove Press, 1953.

[BH09] D. Bohus and E. Horvitz. Dialog in the open world: platform and applications. In *ICMI-MLMI '09: Proceedings of the 2009 international conference on Multimodal interfaces*, pages 31–38, New York, NY, USA, 2009. ACM.

[Bio02] Bioware. Newerwinter nights. http://nwn.bioware.com/, 2002.

[BJJ+04] O. Bangsø, O. G. Jensen, F. V. Jensen, P. B. Andersen, and T. Kocka. Non-linear interactive storytelling using object-oriented bayesian networks. In *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, 2004.

[BK07] H. Barber and D. Kudenko. Dynamic generation of dilemma-based interactive narratives. In *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment conference (AIIDE)*, 2007.

[Bra87] M. Bratman. *Intentions, Plans, and Practical Reason.* Harvard University Press, Cambridge, 1987.

[Byl94] T. Bylander. The computational complexity of propositional STRIPS planning. *Artificial Intelligence*, 69(1-2):165–204, 1994.

[Cat46] R. B. Cattell. *The Description and Measurement of Personality.* New York: Harcourt, Brace, & World, 1946.

[Cat65] R. B. Cattell. *The Scientific Analyses of Personality.* Great Britain: Penguin Books, 1965.

[CCB+06] P. Cairns, A. Cox, N. Berthouze, S. Dhoparee, and C. Jennett. Quantifying the experience of immersion in games. In *Proceedings of the Cognitive Science of Games and Gameplay workshop*, Vancouver, July 2006.

[CD00] J. Cheng and M. J. Druzdzel. AIS-BN: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks. *Journal of Artificial Intelligence Research*, 13:13–155, 2000.

[CL90] P.R. Cohen and H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(2-3):213–261, 1990.

[Cof01] B. Coffin. *Rifts: Game Master Guide*. Palladium Books Inc, September 2001.

[Coo90] G. F. Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 42(2-3):393 – 405, 1990.

[Cra03] C. Crawford. *Chris Crawford on Game Design*. New Riders, 2003.

[Cra04] C. Crawford. *Chris Crawford on Interactive Storytelling*. New Riders, 2004.

[Cul77a] R. E. Cullingford. Controlling inference in story understanding. In *Proceedings of the 5th international joint conference on Artificial intelligence*, pages 17–17, San Francisco, CA, USA, 1977. Morgan Kaufmann Publishers Inc.

[Cul77b] R. E. Cullingford. *Organizing World Knowledge for Story Understanding by Computer*. PhD thesis, Department of Engineering and Applied Science , Yale University, New Haven, Connecticut, 1977.

[Das08] M. Dastani. 2apl:a practical agent programming language. In *Autonomous Agents and Multi-Agent Systems*, 2008.

[Dic12] Oxford English Dictionary. Online, Oxford University Press, 2012.

[DL93] P. Dagum and M. Luby. Approximating probabilistic inference in bayesian belief networks is np-hard. *Artificial Intelligence*, 60(1):141–153, 1993.

[DSL07a] DSL. Genie modeling environment for graphical probabilistic model decision systems laboratory, university of pittsburgh, 2007.

[DSL07b] DSL. SMILE, reasoning engine for graphical probabilistic model, 2007.

 190


[EI19] DONTNOD Entertainment and Focus Home Interactive. Vampyr, 2019.

[Ent04] Blizzard Entertainment. World of warcraft, 2004.

[Ent09] Sony Computer Entertainment. Heavy rain, 2009.

[Ess76] M. Esslin. *An Anatomy of Drama*. Sphere Books, London, 1976.

[Fai04] C. R. Fairclough. *Story Games and the OPIATE System*. PhD thesis, Department of Computer Science, University of Dublin, Trinity College, October 2004.

[FBAP08] R. Figueiredo, A. Brisson, R. Aylett, and A. Paiva. Emergent stories facilitated. In *ICIDS*, pages 218–229. Springer, 2008.

[FC03] C. Fairclough and P. Cunningham. A multiplayer case based story engine. In *GAME-ON*, pages 41–46, 2003.

[FC04] C. R. Fairclough and P. Cunningham. AI structuralist storytelling in computer games. *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, 2004.

[FDA+07] R. Figueiredo, J. Dias, R. Aylett, S. Louchart, and A. Paiva. Shaping emergent narratives for a pedagogical application. Technical report, eCIRCUS, Edinburgh, UK, 2007.

[FF94] R. M. Fung and B. D. Favero. Backward simulation in bayesian networks. In *UAI*, pages 227–234, 1994.

[FPA07] R. Figueiredo, A. Paiva, and R. Aylett. Facilitating the emergence of educational stories - using emergent stories for pedagogical proposes. In *NLE Workshop in AIED*, 2007.

[Fre63] G. Freytag. *Technique of the Drama*. Benjamin Blom, 1863.

[Fre92] R. A. Freeman. The art of the detective story. In *The Art of the Mystery Story*, pages 7–18. Carroll & Graf Publishers, New York, 1992. First published, 1924.

[FT91] D. Fudenberg and J. Tirole. *Game Theory*. The MIT Press, 1991.

[Gam03] CCP Games. Eve online, 2003.

[GD07] C. Glymour and D. Danks. Reasons as causes in bayesian epistemology. *Journal of Philosophy*, 104(9):464–474, 2007.

[Gol93] L. R. Goldberg. The structure of phenotypic personality traits. *American Psychologist*, 48:26–34, 1993.

[Har67] J. C. Harsanyi. Games with incomplete information played by “bayesian" players, I-III. part I. the basic model. *Management Science*, 14(3):159–182, 1967.

[Har68a] J. C. Harsanyi. Games with incomplete information played by “bayesian" players, I-III. part II. bayesian equilibrium points. *Management Science*, 14(5):320–334, 1968.

[Har68b] J. C. Harsanyi. Games with incomplete information played by “bayesian" players, I-III. part III. the basic probability distribution of the game. *Management Science*, 14(7):486–502, 1968.

[Har95] J. C. Harsanyi. Games with incomplete information. *The American Economic Review*, 85(3):291–303, 1995.

[HD96] C. Huang and A. Darwiche. Inference in belief networks: A procedural guide. *International Journal of Approximate Reasoning*, 15:225–263, 1996.

[Hen88] M. Henrion. Propagating uncertainty in bayesian networks by probabilistic logic sampling. In J.F. Lemmer and L.N. Kanal, editors, *Uncertainty in Artificial Intelligence*, volume 2, pages 149–163. North Holland, 1988.

[HHL+92] S. H. Heap, M. Hollis, B. Lyons, R. Sugden, and A. Weale. *The Theory of Choice*. Blackwell, 1992.

[HP99] E. Horvitz and T. Paek. A computational architecture for conversation. In *UM '99: Proceedings of the seventh international conference on User modeling*, pages 201–210. Springer-Verlag New York, Inc., 1999.

[HRK+08] J. Hurliman, J. Radford, L. Khalifa, M. Cortez, et al. LibOpenMetaverse. `https://github.com/openmetaversefoundation/libopenmetaverse`, last accessed on 17.01.2021, 2008.

[HRSB+94] B. Hayes-Roth, E. Sincoff, L. Brownston, R. Huard, and B. Lent. Directed improvisation. Technical report, Stanford Knowledge Systems, 1994. KSL-94-61.

[HRvGH97] B. Hayes-Roth, R. van Gent, and D. Huber. Acting in character. In R. Trappl and P. Petta, editors, *Creating Personalities for Synthetic Actors*. Springer-Verlag, Berlin, 1997.

[Imp21a] Improbable. SpatialOS. `https://improbable.io/games`, last accessed February 2021, 2021.

[Imp21b] Improbable. What is SpatialOS. `https://documentation.improbable.io/spatialos-overview/docs/what-is-spatialos`, last accessed February 2021, 2021.

[Inc00] Ion Storm Inc. Deus ex, 2000.

[Inc17] VRChat Inc. VRChat Inc., 2017.

[Jag01] Jagex. RuneScape, 2001.

[Jen01] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer-Verlag, 2001.

[Jen04] F. V. Jensen. Personal communication, November 2004.

[Jho79] K. Jhonstone. *Impro: Improvisation and the Theatre*. Faber and Faber, 1979.

[Joy22] J. Joyce. *Ulysses*. Sylvia Beach, Paris, 1922.

[KM03] D. Koller and B. Milch. Multi-Agent Influence Diagrams for representing and solving games. *Games and Economic Behavior*, 45(1):181–221, 2003. Full version of paper in IJCAI '03.

[Kno92] R. A. Knox. Detective story decalogue. In *The Art of the Mystery Story*, pages 194–197. Carroll & Graf Publishers, New York, 1992. First published, London: Faber; New York:Liveright, 1929.

[KP97] D. Koller and A. Pfeffer. Object-oriented bayesian networks. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pages 302–313, San Francisco, 1997.

[LA07] S. Louchart and R. Aylett. Building synthetic actors for interactive dramas. In *Proceedings of the AAAI Fall Symposium on Intelligent Narrative Technologies*, pages 63–71, November 2007.

[Lab03] Linden Lab. Second life. `https://secondlife.com/` last accessed on 17.01.2021, 2003.

[LAD07a] S. Louchart, R. Aylett, and J. Dias. Double appraisal for synthetic characters. In *International Conference Intelligent Virtual Agents*. Springer, 2007.

[LAD+07b] S. Louchart, R. Aylett, J. Dias, R. Figueiredo, M. Kriegel, and A. Paiva. Authoring emergent narrative-based games. *Journal of Game Development*, 3(1), 2007.

[LADP05] S. Louchart, R. Aylett, J. Dias, and A. Paiva. Unscripted narrative for affectively driven characters. In *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment conference (AIIDE)*, pages 81–86. AAAI Press, 2005.

[LD97] Y. Lin and M. Druzdzel. Computational advantages of relevance reasoning in bayesian belief networks. In *Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence (UAI-97)*, pages 342–35, San Francisco, CA, 1997. Morgan Kaufmann.

[LDAP08] M. Y. Lim, J. Dias, R. Aylett, and A. Paiva. Improving adaptiveness in autonomous characters. In *IVA*, pages 348–355. Springer, 2008.

[LDAP09] M. Y. Lim, J. Dias, R. Aylett, and A. Paiva. Intelligent NPCs for educational role play game. In *AGS*, pages 107–118. Springer, 2009.

[Leh05] F. Lehár. *The Merry Widow*. Theater an der Wien, 1905.

[LLC02] Bethesda Softworks LLC. Fallout 3, 2002.

[LLC18] Bethesda Softworks LLC. The elder scrolls, 1994-2018.

[LSP+19] B. Liew, M. Scutari, A. Peolsson, G. Peterson, M. Ludvigsson, and D. Falla. Investigating the causal mechanisms of symptom recovery in chronic whiplash-associated disorders using bayesian networks. *The Clinical journal of pain*, 35(8):647–655, 2019.

[LWHS07] D. A. Lagnado, M. R. Waldmann, Y. Hagmayer, and S. A. Sloman. *Causal Learning: Psychology, Philosophy, and Computation*, chapter Beyond Covariation: Cues to Causal Structure. Oxford University Press, USA, 1 edition, March 2007. A. Gopnik, and L. Schultz (Eds.).

[Mac04] B. MacNamee. *Proactive Persistent Agents*. PhD thesis, University of Dublin, Trinity College, Dublin, Ireland, January 2004.

[Mag05] B. Magerko. Story representation and interactive drama. In *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment conference (AIIDE)*, 2005.

[Mag06] B. Magerko. *Player Modeling in the Interactive Drama Architecture*. PhD thesis, The Department of Computer Science and Engineering, University of Michigan, 2006.

[Mat02] M. Mateas. *Interactive Drama, Art, and Artificial Intelligence*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, December 2002. Technical Report CMU-CS-02-206.

[MC03] B. MacNamee and P. Cunningham. Creating socially interactive non player characters: The $\mu$-sic system. *International Journal of Intelligent Games and Simulation*, 2(1):28–35, 2003.

[MDC03] B. MacNamee, S. Dobbyn, and P. Cunningham. Simulating virtual humans across diverse situations. In *In Proceedings of Intelligent Virtual Agents '03*, pages 159–163, 2003.

[Mee81] J. Meehan. Tale-spin. In *In Inside Computer Understanding: Five Programs Plus Miniatures*. Routledge, 1981.

[ML04] B. Magerko and J. E. Laird. Mediating the tension between plot and interaction. In *Proceedings of the AAAI Workshop Series: Challenges in Game Artificial Intelligence*, pages 108–112, 2004.

[ML06] B. W. Mott and J. C. Lester. U-director: a decision-theoretic narrative planning architecture for storytelling environments. *AAMAS*, pages 977–984, 2006.

[MLA+04] B. Magerko, J. E. Laird, M. Assanie, A. Kerfoot, and D. Stokes. AI characters and directors for interactive computer games. In *Proceedings of the Innovative Applications of Artificial Intelligence Conference (IAAI)*, pages 877–883, 2004.

[MPB01] I. Machado, A. Paiva, and P. Brna. Real characters in virtual stories. In *International Conference on Virtual Storytelling*, pages 127–134. Springer, 2001.

[MPG00] C. Martinho, A. Paiva, and M. R. Gomes. Emotions for a motion: Rapid development of believable pathematic agents in intelligent virtual environments. *Applied Artificial Intelligence*, 14(1):33–68, 2000.

[MPP00] I. Machado, R. Prada, and A. Paiva. Bringing drama into a virtual stage. In *International Conference on Collaborative Virtual Environments*. ACM, 2000.

[MPP01] I. Machado, A. Paiva, and R. Prada. Is the wolf angry or ... just hungry? In *Agents*, pages 370–376, 2001.

[MS05a] M. Mateas and A. Stern. Build it to understand it: Ludology meets narratoloy in game design space. In *Proceedings of the Digital Interactive Games Research Association Conference*, Vancouver B.C., June 2005. Included in the Selected Papers volume.

[MS05b] M. Mateas and A. Stern. Structuring content in the facade interactive drama architecture. In *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment conference (AIIDE)*, 2005.

[Nas51] J. Nash. Non-cooperative games. *Annals of Mathematics*, 54(2):286–295, September 1951.

[PC07] D. Pizzi and M. Cavazza. Affective storytelling based on characters' feelings. In *Proceedings of the AAAI Fall Symposium on Intelligent Narrative Technologies*, Arlington, Virginia, November 2007.

[PCLC07] D. Pizzi, F. Charles, J. Lugrin, and M. Cavazza. Interactive storytelling with literary feelings. In *Proceedings of the Second International Conference on Affective Computing and Intelligent Interaction (ACII)*, Lisbon, Portugal, September 2007.

[PDS+04] A. Paiva, J. Dias, D. Sobral, R. Aylett, P. Sobreperez, S. Woods, C. Zoll, and L. E. Hall. Caring for agents and agents that care: Building empathic relations with synthetic agents. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 194–201. IEEE Computer Society, 2004.

[PDS+05] A. Paiva, J. Dias, D. Sobral, R. Aylett, S. Woods, L. E. Hall, and C. Zoll. Learning by feeling: Evoking empathy with synthetic characters. *Applied Artificial Intelligence*, 19(3-4):235–266, 2005.

[Pea86] J. Pearl. Fusion, propagation, and structuring in belief networks. *Artif. Intell.*, 29:241–288, September 1986.

[PMP00] R. Prada, I. Machado, and A. Paiva. Teatrix: Virtual environment for story creation. In *Intelligent Tutoring Systems*, pages 464–473. Springer, 2000.

[PMP01a] A. Paiva, I. Machado, and R. Prada. The child behind the character. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 31(5):361–368, 2001.

[PMP01b] A. Paiva, I. Machado, and R. Prada. Heroes, villians, magicians, ...: dramatis personae in a virtual story creation environment. In *IUI*, pages 129–136, 2001.

[Pol21] G. Polti. *The Thirty-Six Dramatic Situations*. O. Franklin and J.K. Reeve, 1921.

[PPM+04] A. Paiva, R. Prada, I. Machado, C. Martinho, M. Vala, and A. Silva. Life-like characters. tools, affective functions and applications. In *Playing with Agents - Agents in Social and Dramatic Games*, pages 361–376. Springer, 2004.

[PPMG02] R. Prada, A. Paiva, I. Machado, and C. Gouveia. “You cannot use my broom! I'm the witch, you're the prince": Collaboration in a virtual dramatic game. In *Intelligent Tutoring Systems*, pages 913–922. Springer, 2002.

[Pri03] G. Prince. *A Dictionary of Narratology*. University of Nebraska Press, 2003.

[Pro68] V. A. Propp. *Morphology of the Folktale*. University of Texas Press, 2 edition, 1968.

[RC03] S. Rothmann and E. P. Coetzer. The big five personality dimensions and job performance. *SA Journal of Industrial Psychology*, 29(1), 2003.

[RG91] A.S. Rao and M.P. Georgeff. Modeling rational agents within a bdi-architecture. In *In Proceedings of the Second International Knowledge Representation Conference*, 1991.

[RHR96] D. Rousseau and B. Hayes-Roth. Personality in synthetic agents. Technical report, Stanford Knowledge Systems, 1996. KSL-96-21.

[RHR97] D. Rousseau and B. Hayes-Roth. Interacting with personality-rich characters. Technical report, Stanford Knowledge Systems, 1997. KSL-97-06.

[RHR98] D. Rousseau and B. Hayes-Roth. A social-psychological model for synthetic actors. In *Proceedings of the International Conference on Autonomous Agents (Agents'98)*, pages 165–172, 1998.

[Rie04] Eric Ries. IMVU virtual reality. `https://secure.imvu.com` last accessed on 17.01.21, 2004.

[RS06a] M. O. Riedl and A. Stern. Believable agents and intelligent scenario direction for social and cultural leadership training. In *Proceedings of the Conference on Behavior Representation in Modeling and Simulation*, 2006.

[RS06b] M. O. Riedl and A. Stern. Believable agents and intelligent story adaptation for interactive storytelling. In *Proceedings of the Technologies for Interactive Digital Storytelling and Entertainment*, 2006.

[RS06c] M. O. Riedl and A. Stern. Failing believably: Toward drama management with autonomous actors in interactive narratives. In *Proceedings of the Technologies for Interactive Digital Storytelling and Entertainment*, 2006.

[RSY03] M. O. Riedl, C. Saretto, and R. M. Young. Managing interaction between users and agents in a multi-agent storytelling environment. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2003.

[RY06] M. O. Riedl and R. M. Young. From linear story generation to branching story graphs. *IEEE Computer Graphics and Applications*, 26(3), 2006. Special Issue on Interactive Narrative.

[SA77] R . C. Schank and R . P . Abelson. *Scripts, plans, goals and understanding.* Lawrence Erlbaum Press, 1977.

[SAR12] N. Szilas, N. Axelrad, and U. Richle. Propositions for innovative forms of digital interactive storytelling based on narrative theories and practices. *Transaction on Edutainment*, LNCS 7145:161–179, 2012.

[SBB+90] R. Schank, M. Brand, R. Burke, E. Domeshek, D. Edelson, W. Ferguson, M. Freed, M. Jona, B. Krulwich, E. Ohmaye, R. Osgood, and L. Pryor. Towards a general content theory of indices, 1990.

[Sch06] S. Schiesel. Dungeon masters in cyberspace. *The New York Times*, 2006.

[SDM09] M. Sindlar, M. Dastani, and J. Meyer. BDI-based development of virtual characters with a theory of mind. In Z. Ruttkay, M. Kipp, A. Nijholt, and H. Vilhjálmsson, editors, *Intelligent Virtual Agents*, volume 5773 of *Lecture Notes in Computer Science*, pages 34–41. Springer Berlin / Heidelberg, 2009.

[SER16] N. Szilas, S. Estupiñán, and U. Richle. Qualifying and quantifying interestingness in dramatic situations. In F. Nack and A. Gordon, editors, *Proceedings of the Interactive Storytelling. ICIDS 2016. Lecture Notes in Computer Science*. Springer, Cham, 2016.

[SGIR75] R. C. Schank, N. M. Goldman, C. J. Rieger III, and C. K. Riesbeck. Inference and paraphrase by computer. *Journal of the Association for Computer Machinery*, 22(3):309–328, July 1975.

[SGRR73] R. C. Schank, N. Goldman, C. J. Riager, and C. Rissbeck. Margie: memory, analysis, response generation, and inference on english. In *In Proceedings of the Third International Joint Conference on Artificial Intelligence (IJCAI*, pages 255–261, 1973.

[SH10] U. Spierling and S. Hoffmann. Exploring narrative interpretation and adaptation for interactive story creation. In *ICIDS*, pages 50–61, 2010.

[SK87] R. C. Schank and A. Kass. Natural language processing: what's really involved? In *Proceedings of the 1987 workshop on Theoretical issues in natural language processing*, pages 111–115, Stroudsburg, PA, USA, 1987. Association for Computational Linguistics.

[SKe+94] R. C. Schank, A. Kass, C. K. Riesbeck (eds.), M. Brown, and G. Paliouras. Inside case-based explanation, 1994.

[SLB80] R. C. Schank, M. Lebowitz, and L. Birnbaum. An integrated understander. *American Journal of Computational Linguistic*, 6(1):13–30, 1980.

[SM05] N. Szilas and M. Mancini. The control of agents' expressivity in interactive drama. In *Proceedings of the International Conference on Virtual Storytelling, ICVS*, Strasbourg (France), November 2005.

[SM10] M. Si and S. C. Marsella. Modeling rich characters in interactive narrative games. In *GAMEON-ASIA*, Shanghai, China, 2010.

[SMP05a] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: An architecture for interactive pedagogical drama. In *International Conference on Artificial Intelligence in Education*, Amsterdam, 2005.

[SMP05b] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: Using multi-agent fitting to craft interactive drama. In *International Conference on Autonomous Agents and Multi Agent Systems*, 2005.

[SMP06a] M. Si, S. C. Marsella, and D. V. Pynadath. Social norms models in thespian: Using decision theoretical framework for interactive dramas. In *Artificial*

*Intelligence and the Simulation of Behaviour. AISB Symposium, (AISB)*, 2006.

[SMP06b] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: Modeling socially normative behavior in a decision-theoretic framework. In *6th International Conference on Intelligent Virtual Agents*, Marina del Rey, CA, 2006.

[SMP07] M. Si, S. C. Marsella, and D. V. Pynadath. Proactive authoring for interactive drama: An author's assistant. In *7th International Conference on Intelligent Virtual Agents, (IVA)*, Paris, France, 2007.

[SMP08a] M. Si, S. C. Marsella, and D. V. Pynadath. Integrating plot-centric and character-centric processes for authoring interactive drama. In *4th Artificial Intelligence for Interactive Digital Entertainment Conference, (AIIDE)*, 2008.

[SMP08b] M. Si, S. C. Marsella, and D. V. Pynadath. Interactive drama authoring with plot and character: An intelligent system that fosters creativity. In *Proceedings of the AAAI Spring Symposium on Creative Intelligent Systems*, Palo Alto, California, 2008.

[SMP08c] M. Si, S. C. Marsella, and D. V. Pynadath. Modeling appraisal in theory of mind reasoning. In *8th International Conference on Intelligent Virtual Agents, (IVA)*, Japan, 2008.

[SMP09a] M. Si, S. C. Marsella, and D. V. Pynadath. Directorial control in a decision-theoretic framework for interactive narrative. In *Proceedings of International Conference on Interactive Digital Storytelling*, Guimarães, Portugal, 2009.

[SMP09b] M. Si, S. C. Marsella, and D. V. Pynadath. Modeling appraisal in theory of mind reasoning. *Journal of Agents and Multi-Agent Systems*, 20(1):14–31, 2009.

[SMP10] M. Si, S. C. Marsella, and D. V. Pynadath. Evaluating directorial control in a character-centric interactive narrative framework. In *International Conference on Autonomous Agents and Multi Agent Systems, (AAMAS)*, Toronto, Canada, 2010.

[SMR03] N. Szilas, O. Marty, and J. Réty. Authoring highly generative interactive drama. In *Proceedings of the International Conference on Virtual Storytelling*, pages 20–21, Toulouse, France, November 2003.

[SP90] R. D. Shachter and M. A. Peot. Simulation approaches to general probabilistic inference on belief networks. In *Proceedings of the Fifth Annual Conference on Uncertainty in Artificial Intelligence*, UAI '89, pages 221–234, Amsterdam, The Netherlands, The Netherlands, 1990. North-Holland Publishing Co.

[Spi07] U. Spierling. Adding aspects of "implicit creation" to the authoring process in interactive storytelling. In *International Conference on Virtual Storytelling*, pages 13–25, 2007.

[Spi09] U. Spierling. Models for interactive narrative actions. In *In Proceedings of the Interactive Entertainment*, Sydney, Australia, 2009. ACM Digital Library.

[SR04] N. Szilas and J.-H. Rety. Minimal structures for stories. In *Proceedings of the ACM workshop Story representation, Mechanism and Context*, New York, October 2004.

[SRP12] N. Szilas, U. Richle, and P. Petta. Performative structures for interactive narrative: An authored-centred approach. Technical report, The Austrian Research Institute for Artificial Intelligence (OFAI), Universite De Geneve, 2012.

[SS09] U. Spierling and N. Szilas. Authoring issues beyond tools. In *ICIDS*, pages 50–61, 2009.

[Stu16] Cryptic Studios. Neverwinter, 2013-2016.

[SV07] I. Swartjes and J. Vromen. Emergent story generation: Lessons from improvisational theater. In *Proceedings of the AAAI Fall Symposium on Intelligent Narrative Technologies*, Arlington, Virginia, November 2007.

[SVB07] I. Swartjes, J. Vromen, and N. Bloom. Narrative inspiration: Using case based problem solving to support emergent story generation. In *Proceedings of the International Joint Workshop on Computational Creativity*, Goldsmiths, University of London, June 2007.

[SWM06] U. Spierling, S. A. Weiß, and W. Müller. Towards accessible authoring tools for interactive storytelling. In *TIDSE*, pages 169–180, 2006.

[SWV$^{+}$09] M. Sapouna, D. Wolke, N. Vannini, S. Watson, S. Woods, W. Schneider, S. Enz, L. Hall, A. Paiva, E. Andre, K. Dautenhahn, and R. Aylett. Virtual learning intervention to reduce bullying victimization in primary school: a controlled trial. *Journal of Child Psychology and Psychiatry*, 2009.

[Szi01] N. Szilas. A new approach to interactive drama: From intelligent characters to an intelligent virtual narrator. In *Proceedings of the AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment*, Paris (France), 2001.

[Szi02] N. Szilas. Structural models for interactive drama. In *Proceedings of the of the 2nd International Conference on Computational Semiotics for Games and New Media*, Augsburg (Allemagne), September 2002.

[Szi03] N. Szilas. Idtension: a narrative engine for interactive drama. In *Proceedings of the International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, pages 24–26, Darmstadt, Germany, March 2003.

[Szi04] N. Szilas. Stepping into the interactive drama. In *Proceedings of the International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, pages 14–25, Darmstadt, Germany, June 2004. Springer-Verlag.

[Szi05] N. Szilas. The future of interactive drama. In *Proceedings of the Australasian Conference on Interactive Entertainment*, Sydney, November 2005.

[Szi07a] N. Szilas. A computational model of an intelligent narrator for interactive narratives. *Applied Artificial Intelligence*, 21(8):753 – 801, 2007.

[Szi07b] N. Szilas. Towards an author friendly behaviour engine. In *Proceedings of the International Conference on Virtual Storytelling*, Strasbourg (France), December 2007.

[Szi08] N. Szilas. IDtension - highly interactive drama. In *Proceedings of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference*, Switzerland, 2008.

[Szi10] N. Szilas. Requirements for computational models of interactive narrative. In *Proceedings of the Fall Symposium on Computational Models of Narrative, AAAI*, Arlington, VA, USA, November 2010.

[TG01] J. B. Tenenbaum and T. L. Griffiths. Structure learning in human causal induction. *Advances in Neural Information Processing Systems*, 13:59–65, 2001.

[Tod77] T. Todorov. *The Poetics of Prose*, chapter The Typology of Detective Fiction, pages 42–53. Basil Blackwell, Oxford, 1977.

[TS04] A. S. Tanenbaum and M. van Steen. *Distributed Systems Principles and Paradigms*. Pearson Education, 2004.

[Ubi09] Ubisoft. Assassin's creed II, 2009.

[Ubi18] Ubisoft. Assassin's creed series, 2007-2018.

[Van92] S. S. Van Dine. Twenty rules for writing detective stories. In *The Art of the Mystery Story*, pages 189–194. Carroll & Graf Publishers, New York, 1992. First published, American Magazine, September 1928.

[VSPA07] M. Vala, P. Sequeira, A. Paiva, and R. Aylett. FearNot! demo: a virtual environment with synthetic characters to help bullying. In *AAMAS*, page 271. IFAAMAS, 2007.

[Wik21] Wikipedia. Virtual reality. `https://en.wikipedia.org/wiki/Virtual_reality`, last accessed on 17.01.21, 2021.

[Wor09] GSC Game World. S.T.A.L.K.E.R, 2009.

[YD03] C. Yuan and M. J. Druzdzel. An importance sampling algorithm based on evidence pre-propagation. In *In Proceedings of the Nineteenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 624–631. Morgan Kaufmann Publishers, 2003.

[YR03] R. M. Young and M. O. Riedl. Towards an architecture for intelligent control of narrative in interactive virtual worlds. In *Proceedings of the International Conference on Intelligent User Interfaces*, January 2003.

[YRB+04] R. M. Young, M. O. Riedl, M. Branly, A. Jhala, R. J. Martin, and C. J. Saretto. An architecture for integrating plan-based behavior generation with interactive game environments. *Journal of Game Development*, 1, 2004.

[ZPPZ04] C. Zhu, Z. Pan, R. Prada, and M. Zhang. Emotion modeling of virtual chinese characters in E-Teatrix. In *7th International Conference on Computer Graphics and Artificial Intelligence*, 2004.

[ZPTP03] C. Zhu, Z. Pan, B. Tang, and R. Prada. Virtual chinese characters and emotion system in E-Teatrix.

In *1st Chinese Conference on Affective Computing and Intelligent Interaction*, 2003.