

**MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UM MODELO DE DIRETOR DE PERSONAGENS
FIGURANTES PARA O SISTEMA DE NARRATIVA
INTERATIVA LOGTELL**

DISSERTAÇÃO DE MESTRADO

Ramon Flores Limberger

Santa Maria, RS, Brasil

2011

**UM MODELO DE DIRETOR DE PERSONAGENS
FIGURANTES PARA SISTEMAS DE NARRATIVA
INTERATIVA LOGTELL**

Ramon Flores Limberger

Dissertação apresentada ao Curso de Mestrado em Computação do Programa de Pós-Graduação em Informática (PPGI), Área de Concentração em Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de

Mestre em Computação

Orientador: Prof. Dr. Cesar Tadeu Pozzer

Santa Maria, RS, Brasil

2011

**Ministério da Educação
Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Informática**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**UM MODELO DE DIRETOR DE PERSONAGENS FIGURANTES PARA
SISTEMAS DE NARRATIVA INTERATIVA LOGTELL**

elaborada por
Ramon Flores Limberger

como requisito parcial para obtenção do grau de
Mestre em Computação

COMISSÃO EXAMINADORA:

Cesar Tadeu Pozzer, Dr.
(Presidente/Orientador)

Bruno Feijó, Dr. (PUC-RIO)

Lisandra Manzoni Fontoura, Dra. (UFSM)

Santa Maria, 14 de setembro de 2011.

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria
Ministério da Educação

UM MODELO DE DIRETOR DE PERSONAGENS FIGURANTES PARA SISTEMAS DE NARRATIVA INTERATIVA LOGTELL

Autor: Ramon Flores Limberger

Orientador: Cesar Tadeu Pozzer

Data e local da defesa: Santa Maria, 14 setembro de 2011.

O objetivo desta dissertação é criar um modelo de Diretor para personagens figurantes em Narrativa Interativa, buscando enriquecer a dramatização e aumentar o realismo das histórias geradas. A atuação dos personagens figurantes se relaciona também ao tempo de duração das histórias interativas. Uma história apenas com eventos principais dificilmente terá uma longa duração. Ao observar filmes e novelas, nota-se que existem uma série de eventos secundários e personagens figurantes para aumentar a dramatização da história. Nos sistemas de Narrativa interativa, atualmente, personagens figurantes é um tema pouco explorado, consequentemente, as histórias não conseguem ter um tempo de duração similares ao que ocorrem em filmes e novelas. Nesta dissertação, é definido um diretor de personagens figurantes para o sistema de Narrativa Interativa *Logtell*, detalhando os problemas e facilidades para inserção dos personagens figurantes nos cenários das histórias, explicando os passos e as soluções encontradas para aumentar o tempo de dramatização e realismo nas histórias.

Palavras-chave: Narrativa Interativa, Personagens Figurantes, Computação Gráfica.

ABSTRACT

Master's Dissertation
Post-Graduate Program in Informatics
Federal University of Santa Maria

A MINOR CHARACTERS DIRECTOR MODEL FOR INTERACTIVE STORYTELLING IN LOGTELL

Author: Ramon Flores Limberger

Advisor: Cesar Tadeu Pozzer

Santa Maria, 14 September, 2011.

The goal of this dissertation is to create a supporting characters director model for Interactive Storytelling, trying to enrich the drama of the generated stories . Supporting characters also relates to the running time of Interactive Storytelling stories. A story with only the major events is unlikely to have a long duration. When looking at films and novels, there are a number of side events and supporting characters to increase the drama of the story. In Interactive Storytelling systems nowadays supporting characters is a subject rarely treated, and thus the stories cannot have a longer duration. In this dissertation it is defined a supporting characters director to the Interactive Storytelling System Logtell, explaining in detail the integration process and what steps that were taken to increase the drama and realism in generated stories.

Keywords: Interactive Storytelling, Supporting characters, Computer Graphics.

LISTA DE FIGURAS

Figura 1. Cena do filme Senhor dos Anéis.....	15
Figura 2. Exemplo de duas multidões simuladas seguindo os passos.....	16
Figura 3. Gráficos de sistema de navegação por vértices.....	17
Figura 4. Cenário e personagens reconstruído de Pompeii.	18
Figura 5. Cenário teste do modelo.	19
Figura 6. Plano para o personagem Ross.	22
Figura 7. Replanejamento após interferência do usuário.	22
Figura 8. Ross indo ao local do diário que foi movido por interferência do usuário.	24
Figura 9. Arquitetura do sistema incluindo o gerador de diálogos.....	25
Figura 10. <i>Screenshot</i> do sistema Façade.....	27
Figura 11. Arquitetura do <i>Logtell</i>	28
Figura 12. Interface do <i>Plot Manager</i> com um exemplo de estória gerada.	29
Figura 13. Exemplo de estória gerada pelo IPG.....	31
Figura 14. Comparação entre o evento “ <i>Fight</i> ” antigo e o atual com a <i>Unity</i>	31
Figura 15. Terrenos criados na <i>Unity</i> pelo <i>Logtell</i>	32
Figura 16. Ilustração da geração de <i>Waypoints</i>	34
Figura 17. Grupo de personagens se locomovendo no cenário.	36
Figura 18. Ilustração das rotas no cenário.	38
Figura 19. Modelo do Personagem Figurante com seus <i>colliders</i>	39
Figura 20. Modelo final do Personagem Figurante e do Personagem Principal com seus <i>colliders</i>	40
Figura 21. Personagens Figurantes Estáticos nos cenários.	41
Figura 22. Personagens Figurantes de segundo nível.....	42
Figura 23. Formação da plateia na luta.	43
Figura 24. Animação dos figurantes durante a luta.....	44
Figura 25. Diferentes reações dos figurantes durante a luta.....	44
Figura 26. Nova câmera no Evento “GO”.....	45
Figura 27. Diferentes tomadas de câmera apresentando a região.	46
Figura 28. Diferentes tomadas de câmera com as emoções.	48
Figura 29. Diálogo do Personagem Figurante de segundo nível e Personagem Principal.	49
Figura 30. Arquitetura do sistema <i>Logtell</i>	50

Figura 31. Interface da Unity.	52
Figura 32. Versão antiga do Logtell e a nova versão com o Diretor de Figurantes.	56
Figura 33. Comparação entre as versões do Logtell.	58
Figura 34. Exemplo de sequência de estória..	60

LISTA DE ABREVIATURAS E SIGLAS

CH1	<i>Character 1</i>
CH2	<i>Character 2</i>
EM	<i>Emotion</i>
HTN	<i>Hierarchical Task Network</i>
IPG	<i>Interactive Plot Generator</i>
PF	Personagens Figurantes
PL	<i>Place</i>
VIC	<i>Victim</i>
VIL	<i>Vilain</i>

SUMÁRIO

1. INTRODUÇÃO.....	10
1.1. Objetivos.....	12
1.2. Contribuições	12
1.3. Estrutura	13
2. REVISÃO BIBLIOGRÁFICA	14
2.1. Simulação de multidões.....	14
2.2. Sistemas de narrativa interativa	21
2.2.1. <i>Hierarchical Task Network</i> (HTN).....	21
2.2.2. Façade	26
3. LOGTELL	28
4. DIRETOR DE FIGURANTES	33
4.1. Geração de Waypoints	33
4.2. Geração de personagens figurantes	35
4.3. Personagens Figurantes Estáticos	40
4.4. Personagens Figurantes de segundo nível	41
4.5. Fight Scene	42
4.6. Manipulação de câmeras	44
4.7. Emoção dos Figurantes	46
4.8. Arquitetura com <i>Logtell</i>	49
5. IMPLEMENTAÇÃO.....	52
6. RESULTADOS	55
7. CONCLUSÃO	61
7.1. Trabalhos Futuros	62
BIBLIOGRAFIA.....	64

1. INTRODUÇÃO

A área de entretenimento ganhou, nos últimos anos, uma forte ênfase em pesquisas científicas. Com a evolução tecnológica e com a convergência digital, novas mídias podem ser desenvolvidas, facilitando assim a criação de jogos e histórias, permitindo também inserir novos paradigmas para manipular histórias, considerando a interação dos usuários (MILLER, 2004).

A Narrativa Interativa é uma técnica de entretenimento que combina a forma tradicional de contar ou desenvolver histórias inserindo a interatividade do usuário/expectador à dramatização que se desenvolve. Desenvolver uma narrativa para esse ambiente deve permitir que a história se desenvolva de uma maneira autônoma, mas, ao mesmo tempo, permitir que possa ser modificada em sua execução com a intervenção do espectador. Permite assim que o espectador, de uma forma subjetiva, insira na história que está se desenvolvendo as suas experiências, valores e significados pessoais, combinando possibilidades e assistindo assim a uma história exclusiva (BRAUN et al., 2002).

Assim, a Narrativa Interativa vem a modificar histórias tradicionais proporcionando maior interação entre os espectadores e os personagens. A Narrativa Interativa se aplica a um vasto campo de entretenimento, desde filmes até desenvolvimento de jogos.

Em um sistema de Narrativa Interativa, é muito importante manter a coerência nas histórias enquanto permite a interação com o usuário. Como é muito difícil manter essa coerência em histórias muito detalhadas com eventos de baixo nível (maior nível de detalhes), as histórias geradas nos sistemas de Narrativa Interativa atuais são mais focadas em eventos de mais alto nível envolvendo apenas eventos principais. Um dos principais problemas com histórias que não possuem eventos secundários é a dificuldade de essa história gerada possuir um maior tempo de duração. Se os filmes/novelas fossem formados apenas com os eventos principais eles não teriam a longa duração que apresentam dificultando assim atrair o público espectador.

No desenvolvimento das pesquisas na área da narrativa interativa, estas podem ser classificadas em três linhas de pesquisa: a geração das histórias, nas quais se usam métodos automatizados para gerar enredos coerentes e diversificados; a interação dos usuários com a

estória, que define a forma como essa interação irá interferir no contexto da estória; e a dramatização das estórias geradas, em que é realizada a visualização dos eventos.

Olhando sistemas de narrativas já criados, observa-se que eles têm em comum a falta de eventos secundários. Segundo Tobias (1993), não é possível criar uma estória interessante sem obstáculos entre os protagonistas e seus objetivos.

Uma estória para ter uma duração significativa de tempo, necessita de diversos incidentes menores. Dessa forma, além dos personagens principais, deve existir todo um elenco de personagens figurantes (PFs) por trás, para dar suporte no desenvolvimento da estória. Os PFs têm diversas funções e finalidades, entre outras eles: podem povoar as ruas de uma cidade enquanto o ator principal está se locomovendo; podem representar um amigo de confiança do personagem principal; ou podem representar um simples motorista de taxi (GALEF, 1993).

Considerando a necessidade de facilitar e melhorar a estória, que está sendo contada, especialmente disponibilizando o recurso da interatividade ao espectador, o uso dos personagens figurantes permite enriquecer o cenário, criar facilidades e/ou obstáculos, apoiar o enredo, indicar caminhos ou rotas a serem seguidas, alongar o tempo de dramatização das estórias. Em resumo, eles têm funções específicas na estória que ajudam a enriquecer o enredo.

Nesta dissertação, é descrito um modelo de diretor de personagens figurantes para um sistema de narrativa interativa, com o objetivo de estender as estórias bem como enriquecer a dramatização. O modelo foi incorporado ao *Logtell* (POZZER, 2005; LIMA, 2010), um sistema baseado em enredo, mas que permite a interferência do usuário. No *Logtell* existe um conjunto de eventos modelados por meio de operações parametrizáveis (andar, raptar, lutar, casar, entre outras). Esse conjunto de eventos é passado para o sistema de dramatização, o qual é encarregado pela apresentação gráfica 3D para o usuário. Maiores detalhes sobre o *Logtell* é visto no capítulo 3. A estória utilizada como exemplo para este trabalho no *Logtell* é baseada em contos de fadas e possui dois heróis, um vilão e uma princesa. No desencadeamento *default* da estória em questão, o vilão rapta a princesa, o herói (Brian) se desloca ao castelo do vilão para combatê-lo e libertar a princesa e, ao final, o herói e a princesa se casam. Deve-se observar que, na versão original do *Logtell*, não existem personagens figurantes, o que torna a dramatização muito determinística e com poucos detalhes e baixa variabilidade.

1.1. Objetivos

O principal objetivo desta dissertação é criar um modelo de diretor de personagens figurantes para, com sua ação, aumentarem a dramaticidade e o realismo de sistemas de narrativa interativa. O Diretor de Figurantes será encarregado de gerar e gerenciar personagens figurantes, em tempo real, durante a dramatização das histórias no ambiente do *Logtell*. Os personagens figurantes irão povoar o cenário de forma que o cenário da história tenha uma maior semelhança com a realidade.

Como exemplo prático, pode-se descrever o povoamento de regiões ou cidades por onde personagens principais se deslocam. Assim, personagens figurantes podem, por exemplo, ocupar varandas das casas, além de se deslocar pelas ruas da região em cenas de deslocamento de um personagem principal. Os Personagens Figurantes também serão encarregados de interagir com os personagens principais em determinadas situações com o intuito de aumentar a dramaticidade da história. O diretor de figurantes também deve gerar novas tomadas de câmeras para melhor visualização dos personagens, uma vez que as câmeras atuais do *Logtell* foram criadas numa versão sem personagens figurantes, portanto não conseguem filmar de forma ideal esses personagens.

Embora o diretor de personagens figurantes vá ser aplicado em um modelo de narrativa interativa, ele deve ser capaz de gerar/gerenciar os personagens figurantes em tempo real de forma autônoma, independente do cenário em que será aplicado.

1.2. Contribuições

Esta dissertação objetiva propor um modelo de diretor de personagens figurantes com o intuito de atingir as seguintes contribuições:

- Criação de um modelo de diretor de Personagens Figurantes, capaz de povoar um cenário de forma realista e que valorize a dramaticidade;
- Modelo para que os personagens figurantes possam interagir com os personagens principais com intuito de enriquecer a dramatização e alongar as histórias;
- Aumento do poder de interação do usuário com a história, com o usuário podendo escolher o quão povoado cada região pode ser, assim como o nível de interferência que os Personagens Figurantes irão ter na história.

1.3. Estrutura

A dissertação está estruturada da seguinte forma: O capítulo 2 apresenta uma revisão bibliográfica de trabalhos relacionados ao tema de Personagens Figurantes, primeiramente descrevendo sobre áreas de deslocamento em massa, como simulação de multidões e *Steering Behaviors* e, posteriormente, detalhando alguns sistemas de Narrativa Interativa. O capítulo 3 detalha o sistema de narrativa interativa *Logtell*, no qual o Diretor de Figurantes será aplicado. O capítulo 4 apresenta a estrutura e funcionamento do diretor de personagens figurantes. No capítulo 5, é descrito a implementação do diretor de Personagens Figurantes e no capítulo 6, são apresentados os resultados. O capítulo 7 apresenta as conclusões deste trabalho, assim como propostas de trabalhos futuros.

2. REVISÃO BIBLIOGRÁFICA

Neste capítulo, é feita a revisão bibliográfica da dissertação buscando estudos anteriores já desenvolvidos e que estão relacionados ao trabalho aqui apresentado. A primeira seção revisa a área de simulação de multidões, área que possui uma relação com a inserção dos personagens figurantes nas histórias que se desenvolvem. Depois é realizada uma revisão dos sistemas de narrativa interativa e a ligação que cada um desses sistemas tem com a inserção dos personagens figurantes.

2.1. Simulação de multidões

Embora não seja o objetivo desta dissertação simular multidões, essa área de pesquisa tem diversos aspectos estudados em seus conceitos básicos e que serão detalhados nesta revisão e que podem ser aplicados ao gerenciamento de personagens figurantes. Na apresentação, neste trabalho, a simulação de multidões pode ser dividida em duas principais áreas (ou pode ser tratada sob aspecto visual e de fluxo): no aspecto de realismo comportamental, que normalmente são simples visualizações bidimensionais (como simulação de evacuações), e na alta-qualidade de visualização (como em filmes e jogos de computadores) (THALMANN; MUSSE, 2007).

Além dessas áreas de mais alto-nível, que tratam a multidão como um bloco uniforme com objetivos específicos, os problemas que se apresentam nesses estudos e tratados nessas pesquisas em níveis mais detalhados também são de interesse para personagens figurantes em narrativas interativas. Como exemplos podem ser destacados a criação de *paths* (rotas ou caminhos das multidões) e o tratamento de colisões em cada agente da multidão.

A Figura 1 apresenta um exemplo de simulação de multidões utilizada no filme Senhor dos Anéis. Nessa figura pode ser visualizada a composição uniforme e densa do povoamento da multidão utilizada no apoio à história que se desenvolve.



Figura 1. Cena do filme *Senhor dos Anéis*.
Fonte: Massive Software, 2010

Pettré et al. (2006), em suas pesquisas, propõem um modelo para distribuição de uma multidão bastante populosa, dentro de um cenário qualquer de forma realista. O artigo no qual é apresentado o resultado de sua pesquisa é dividido fundamentalmente em duas partes principais: na estruturação do ambiente, com a geração dos *paths* no cenário; e na simulação da multidão em tempo real. Na estruturação do ambiente, são realizados os seguintes passos (ilustrados na Figura 2):

- **Passo 1 (*Navigation Grid Points*):** Definem as possibilidades de locomoção. Os *grid points* são os pontos no cenário nos quais o personagem pode ultrapassar, ou seja, aqueles que não são obstáculos para locomoção.
- **Passo 2 (*Navigation Grid Mesh*):** É a tentativa de conexão entre um *grid point* e seus quatro vizinhos, que representam a conexão entre dois pontos por pelos quais o personagem pode se deslocar, ou seja, em deslocamentos livres de colisões. Se um *grid point* tem três ou menos vizinhos é deduzido que é um ponto de borda.
- **Passo 3 (*Clearance Map*):** *Clearance* de um *grid point* é a distância entre o *grid point* e o ponto de borda mais próximo.
- **Passo 4 (*Navigation Graph Construction*):** Os vértices são deduzidos partindo dos *grid points* e da *clearance*. De acordo com a definição de *clearances*, são criadas circunferências, cujo centro é o *grid point* considerado, e o raio é a *clearance*. Por meio dessas circunferências, são definidas a forma e direção da navegação dos personagens.

Após estruturar o ambiente da estória que se desenvolve, é possível a simulação, em tempo real, da multidão. Para isso, o usuário define dois *grid points* pelo qual a multidão irá se deslocar. Depois de definidos os *grid points*, é usado o algoritmo *Dijkstra* para encontrar o caminho mais próximo entre esses dois pontos. Na sequência, é utilizado novamente o mesmo algoritmo para se encontrar outros *paths* alternativos. Após encontrado o *path*, é criado um número de *waypoints*, dependendo da precisão estabelecida pelo o usuário. Esses *waypoints* ficam localizados nos portões (*gates*), que são as intersecções entre as circunferências. Um exemplo dessas simulações está apresentado na Figura 2.



Figura 2. Exemplo de duas multidões simuladas seguindo os passos.
Fonte: Pettre et al., 2006.

Embora este artigo tenha diversos conceitos interessantes, sobretudo na geração de *paths* dinâmicos em cenários diferenciados, ele é direcionado à geração de simulações de multidões bastante populosas embora ele apresente detalhes importantes a serem utilizados para os personagens figurantes. Uma expansão deste trabalho é proposto pelos mesmos autores em Maïm et al. (2007).

Maïm et al. (2007), em seu trabalho, propõem reconstruir e povoar a cidade romana de Pompeii, que foi queimada durante a erupção do vulcão *Mount Vesuvius*. A cidade reconstruída foi dividida pelos autores em duas regiões: as regiões novas, onde circulavam a população nobre; e as regiões mais antigas, onde circulava a população mais pobre (como os escravos). Para aumentar as viabilidades, foram criados lugares onde a população poderia entrar, como em lojas e padarias. Para a movimentação da multidão pela cidade foi utilizado o sistema de navegação por vértices (visto no artigo anterior), porém, apresenta como diferença, que neste trabalho é possível observar o comportamento e postura dos agentes. Destaca-se assim que, em circunferências geradas próximas às janelas e portas de uma casa, a população

tem uma reação diferenciada, diminuindo a velocidade e observando a movimentação janelas ou portas. Esses comportamentos estão representados na Figura 3 com as funções específicas como: “olhe para” (em branco); “pare de olhar para” (em preto); e “ponto destino” (em vermelho).

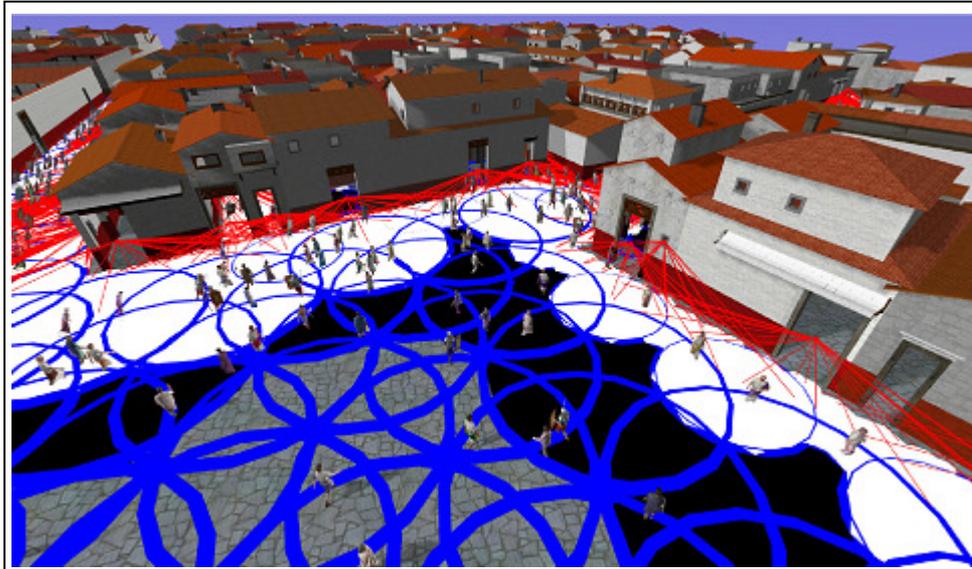


Figura 3. Gráficos de sistema de navegação por vértices.
Fonte: Maïm et al., 2007.

Este artigo é também mais interessante na sua aplicabilidade aos personagens figurantes no que se diz respeito ao objetivo desta dissertação, pois se preocupa mais com o aspecto visual e na simulação de cada agente individualmente ao contrário de apenas criar uma multidão bastante populosa. Essa simulação permite inclusive a criação de personagens específicos (do gênero masculino ou feminino: nobres, plebeus, patrícios, e um legionário), e um cenário com um aspecto visual com maior semelhança ao cenário necessário e utilizado nesta dissertação. Na Figura 4, são representados, em um primeiro momento, a reconstrução do cenário de Pompeii e, ao lado, os personagens utilizados na representação.

Para os sistemas de narrativas interativas, a interação proposta por Maïm et al. (2007) ainda não é o suficiente. Embora no artigo de Maïm et al. (2007) já esteja prevista uma interação maior com os personagens interagindo com portas e janelas, para os objetivos de um sistema de narrativa interativa, os personagens figurantes precisarão interagir com os personagens principais e os usuários, além de permitir maiores detalhes da configuração e ocupação dos cenários.



Figura 4. Cenário e personagens reconstruído de Pompeii.
Fonte: Maïm et al., 2007.

Reynolds, personagem autônomo é um tipo de agente autônomo que tem o propósito de ser utilizado em animações computacionais, como em jogos e em cenas de realidade virtual. Esses agentes podem representar personagens em histórias ou jogos e possuem algumas técnicas

Outra forma para a locomoção de personagens figurantes é pela técnica de *Steering Behaviors*, proposto por Reynolds (1999). Segundo para aprimorar a sua locomoção. Diversos conceitos são propostos por Reynolds para os personagens autônomos, dentre estes, alguns são explicados abaixo:

- *Seek*: Serve para locomover o personagem até uma determinada posição no espaço.
- *Flee*: Contrário de *seek*, o personagem se move em posição oposta a uma determinada posição no espaço.
- *Path Following*: O personagem se locomove dentro de um *path* pré-determinado.
- *Cohesion*: Dá a habilidade ao personagem de aproximar e formar grupos com outros personagens que estão próximos.
- *Alignment*: Dá a habilidade ao personagem de se alinhar com outros personagens (vetor direção).
- *Flocking*: Tem a função de formar “rebanhos e dá-se pela adição das forças de *Separation*, *Alignment* e *Cohesion*. Com essas três simples regras, o “rebanho” se move de uma forma bastante realista.

- Combinando *Behaviors*: Para criar personagens com verdadeiro realismo, faz-se necessária a união de mais de um *steering behavior*.

Também em simulação de multidões, podemos destacar o trabalho de Musse e Thalmann (1997), que descreve um modelo de simulação de multidão levando em consideração as relações entre grupos e o comportamento individual de cada agente. Nessa abordagem, cada agente é tratado como um humano virtual que reage na presença de outros agentes e permitem que estes mudem seus parâmetros de acordo com um novo relacionamento. Cada agente pertence a um determinado grupo. Os agentes possuem diversos parâmetros como: lista de objetivo, lista de interesses e *status* emocional.

Baseado nestes parâmetros, Musse e Thalmann (1997) propõe uma dinâmica com situações de possíveis mudanças onde os agentes podem trocar de grupos, como por exemplo: se a afinidade entre o agente A e o grupo A for menor do que a relação/afinidade entre o agente A e o grupo B, o agente A passa a pertencer ao grupo B e sua lista de objetivos é atualizada de acordo com a lista de objetivos do grupo B.

A Figura 5 apresenta um exemplo de simulação de multidões do modelo proposto por Musse e Thalmann (1997).

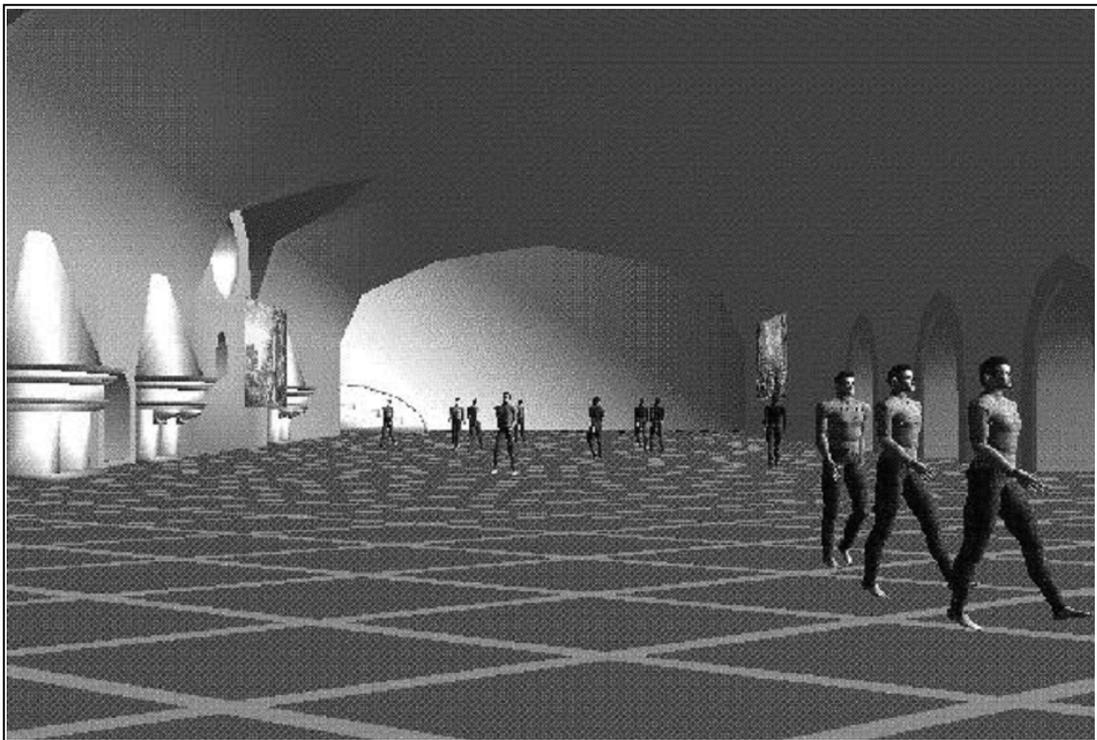


Figura 5. Cenário teste do modelo.
Fonte: Musse e Thalmann, 1997.

Musse e Thalmann (1997) ainda propõem no artigo três tipos de tratamento de colisões entre os agentes da multidão. O primeiro é bem simples e é baseado em equações matemáticas para determinar posições futuras dos agentes e detectar, nesse movimento, as possíveis colisões. Para cada par de agentes, verificam as seguintes perspectivas de colisões após a movimentação:

- Vetores não colineares:
 - Velocidades lineares diferentes: Para o agente com menor velocidade
 - Velocidades lineares iguais: Para um agente escolhido de forma aleatória.
- Vetores colineares:
 - Vetores convergentes: Muda a direção de um agente escolhido de forma aleatória.
 - Vetores não convergentes: Aumenta a velocidade do agente que está na frente.

O segundo tipo de tratamento de colisão analisado por Musse e Thalmann (1997) também é bastante simples. Em vez dos agentes esperarem os outros passarem, como ocorre no primeiro tipo de tratamento, nesse segundo, a colisão é tratada inserindo-se uma mudança de direção e alteração da velocidade de cada agente.

O terceiro e último tipo de tratamento de colisão proposto é chamado de colisão de multirresolução. Neste, dependendo da distância dos agentes para a posição do observador, utiliza-se: a colisão do tipo 2 (para agentes localizados perto da área principal de visualização), colisão do tipo 1 (para distância intermediária), ou ainda nenhum tipo de tratamento de colisão (para distâncias maiores). Esse último tipo de tratamento foi o mais usual em números de ocorrências e não envolve processamento adicional para sua implementação. Esse algoritmo, dessa forma, tem o melhor desempenho, pois tem o melhor tempo de processamento além de possuir um efeito visual satisfatório. Isso decorre porque a grande maioria das colisões ocorre a uma distância que o observador não irá visualizar a colisão entre os agentes e, assim, não sendo prejudiciais para as tomadas do cenário exibidas para o usuário/espectador.

É possível absorver, dos estudos relacionados com a simulação de multidões, diversos aspectos de interesse para inserir os personagens figurantes nas histórias e definir atribuições a serem alocadas ao diretor de personagens figurantes que serão apresentados no decorrer deste trabalho. Outra área de importante contribuição para este trabalho é a de humanos virtuais (GARCIA-ROJAS et al. 2008; McCRAE; JOHN, 1992). Diferentemente da área de simulação de multidões que trata mais a multidão como um todo, a área de humanos virtuais é mais

focada para cada agente individualmente, assemelhando-se mais à proposta deste trabalho que é gerar personagens figurantes realistas capazes de interagir com os personagens principais e a estória.

2.2. Sistemas de narrativa interativa

Nesta seção, serão apresentados alguns dos principais sistemas de narrativa interativa desenvolvidos e que tenham alguma relação com o trabalho aqui desenvolvido. Nestes, é realizada uma análise da forma como os personagens figurantes foram abordados em cada um dos sistemas, buscando subsidiar seu uso nas configurações necessárias para definir as atribuições do diretor de personagens figurantes proposto neste trabalho.

2.2.1. *Hierarchical Task Network* (HTN)

Cavazza et al. (2001) propõe um modelo para *Character-Based Storytelling* baseado em uma *Hierarchical Task Network* (HTN)). Esse modelo consiste no desenvolvimento de um plano de ação para cada personagem (ator) na estória. O HTN é uma árvore de decisão cujos ramos finais (folhas) são ações terminais, e os ramos iniciais são atividades genéricas relacionadas a grupos de personagens.

Na Figura 6, tem-se um exemplo de uma HTN para um personagem específico de uma determinada estória. O artigo de Cavazza propõe gerar uma cena partindo do famoso seriado da NBC *Friends*, no qual o personagem Ross tenta convidar a personagem Rachel para sair. Para alcançar o seu objetivo de sair com Rachel, Ross tem alguns planos pré-definidos (no desenvolvimento da estória) como pedir informação a um amigo de Rachel, ler o diário dela, ou ainda questionar diretamente para ela (Rachel). Cada ação vai depender do seu estado emocional, por exemplo, se Ross é um personagem tímido, ele irá tender a não convidar diretamente Rachel para sair usando assim uma das duas outras alternativas de ação.

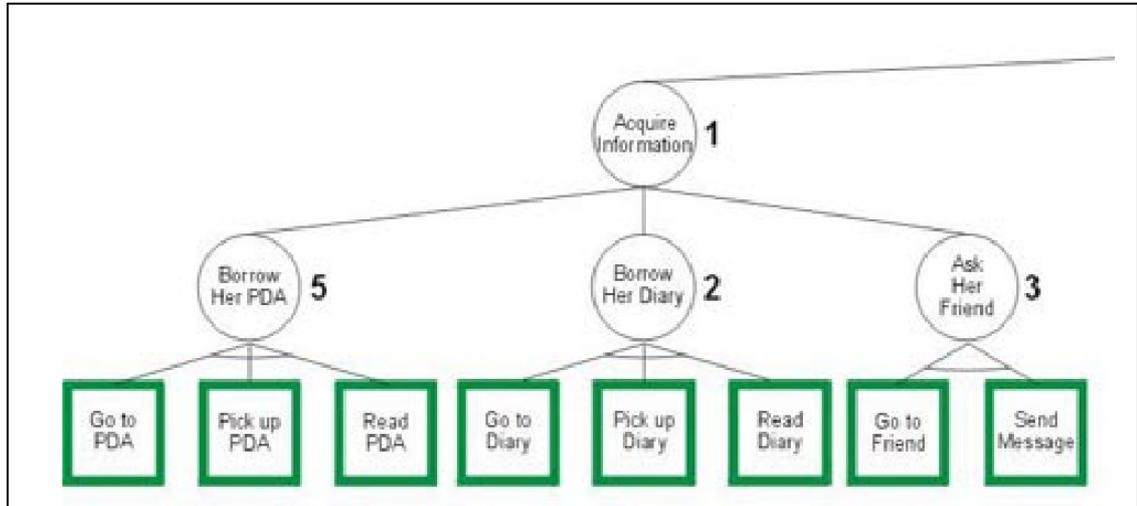


Figura 6. Plano para o personagem Ross.
Fonte: Cavazza et al., 2001.

Nesse modelo, o usuário (espectador) pode interferir diretamente na estória enquanto esta se desenvolve. Um exemplo de interação possível nessa estória é quando o personagem Ross está em busca de informação sobre a personagem Rachel. Se Ross decidir buscar informação pegando o diário de Rachel, o usuário pode escolher por esconder o diário ou forçar Ross a fazer um replanejamento, conforme apresentado na Figura 7.

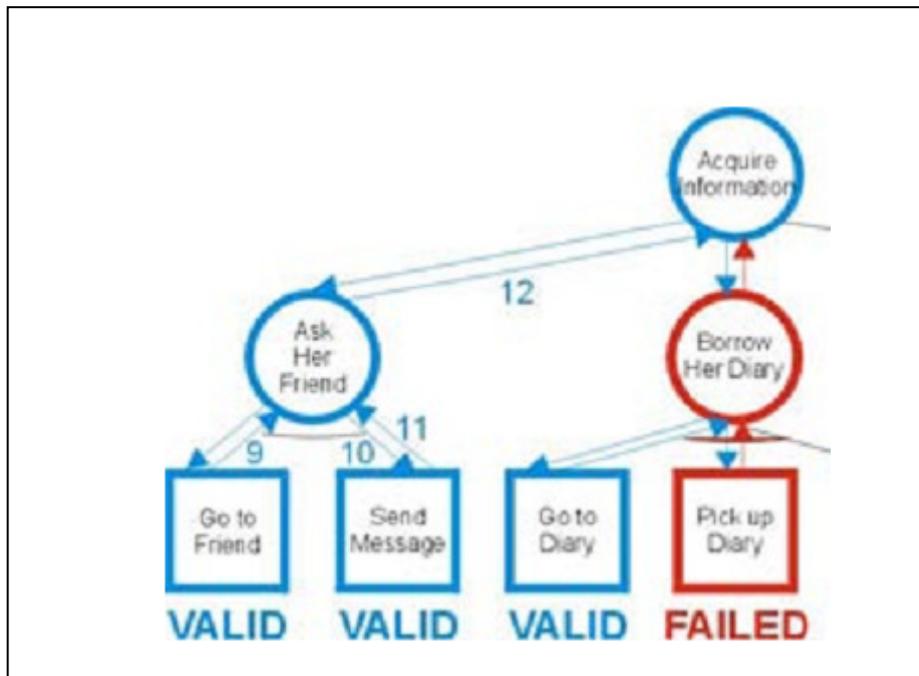


Figura 7. Replanejamento após interferência do usuário.

Fonte: Cavazza et al., 2001.

Cavazza et al. (2001) recomenda ainda neste artigo para importância da história possuir personagens secundários, embora ele não se aprofunde em detalhes sobre esses personagens. Em seu trabalho ele apenas explica que seriam agentes reativos e que estes não teriam uma HTN própria, e eles apenas poderiam influenciar no estado emocional dos personagens principais na cena. No trabalho de Cavazza et al. (2001) foi utilizada a *Unreal Engine*, para a demonstração gráfica das cenas, como demonstrado na Figura 8.



Figura 8. Ross indo ao local do diário que foi movido por interferência do usuário.
Fonte: Cavazza et al., 2001.

Cavazza e Charles (2005) propõem uma extensão do seu trabalho anterior, no qual os personagens, além das HTNs, possuem uma geração de diálogos dinâmicos. O objetivo principal do gerador de diálogos é produzir um diálogo apropriado para uma determinada situação sem esta ter sido gerada pelo sistema de narrativa interativa. Para isso, são utilizados uma matriz de afinidades entre os personagens e um banco de dados com diálogos pré-definidos para determinadas situações.

No trabalho de Cavazza e Charles (2005), a atividade simulada é um convite para uma festa. Na Figura 9, é demonstrado o modelo do gerador de diálogos proposto, que é compreendido por quatro principais componentes:

- i) Uma interface com o HTN de cada personagem assim como a afinidade entre os personagens;
- ii) uma *engine* baseada em regras que controla as opções de geração do modelo;
- iii) um modelo baseado em um gerador semântico, que produz os modelos de geração da narrativa; e
- iv) um modelo para produzir o enunciado final da linguística.

Neste trabalho, é importante destacar, para o desenvolvimento de personagens figurantes, da necessidade de conhecer a forma de atuação dos personagens principais, para

que, com base nesse conhecimento e das ações por eles desenvolvidas, definir as viabilidades de interação dos personagens figurantes com as suas ações.

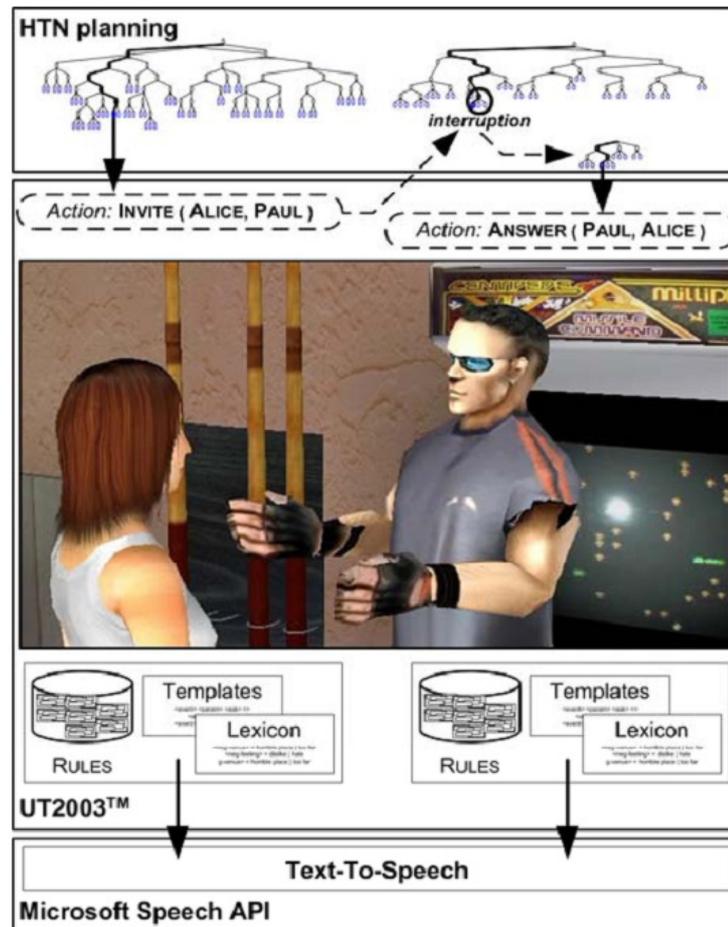


Figura 9. Arquitetura do sistema incluindo o gerador de diálogos.
Fonte: Cavazza e Charles, 2005.

Em Cavazza (2004), é discutido a importância dos personagens figurantes em *Character-Based Storytelling*. Cavazza comenta sobre a necessidade de quebrar uma barreira de 10 minutos de tempo de duração de uma história, preservando sua coerência. Novamente nesse trabalho de Cavazza, são utilizadas as HTNs para os personagens. Os personagens figurantes podem inferir com as ações dos personagens principais, podendo ter atividades como conversar com outros personagens, jogar fliperama e comprar itens de alguma determinada máquina. As HTNs dos personagens figurantes são limitadas, como citado no trabalho de Cavazza (2001). Cavazza conclui nesse trabalho que o acréscimo de personagens figurantes não influencia no tempo total da história e não contribui para elementos significantes do enredo, uma vez que suas ações são restritas a atividades pouco complexas sem objetivos adequados na narrativa.

Dessa forma, os sistemas de narrativa interativa apresentam fundamentações importantes a serem utilizadas neste trabalho, especialmente na definição das funcionalidades e nas viabilidades de interação dos personagens figurantes com os personagens principais das histórias e, dessa forma, facilitar a ampliação do enredo das histórias desenvolvidas e na ampliação das perspectivas de interação do usuário (espectador) com a história que está se desenvolvendo.

2.2.2. Façade

Mateas (2003) propõe um modelo de *Interactive Drama* chamado Façade. Nesse modelo, o usuário interage via mensagens de texto diretamente com os personagens. Neste exemplo, apresenta dois personagens: Trip e Grace, um casal bem-sucedido. O usuário (espectador) interpreta o papel de um amigo de longa data do casal. Na trama, o casal começa a ter desentendimentos e suas ações vão depender da interação com o usuário que, com suas ações, pode interferir diretamente no rumo da história.

O cenário é o apartamento do casal e não possui tratamento com personagens figurantes, uma vez que os únicos personagens da trama são Trip, Grace e o amigo do casal (o usuário). O principal atrativo desse Drama é a grande variedade de situações que podem ocorrer, dependendo da interação com o usuário. A Figura 10 apresenta um *screenshot* mostrando os personagens principais (Trip e Grace), uma mensagem de texto do usuário e o apartamento do casal.

Embora o Façade tenha sido de grande contribuição para a área de narrativas interativas, viabilizando uma grande interação com o usuário e a enorme capacidade desse usuário de interferir no rumo da história, tal modelo não insere personagens figurantes no enredo.

De forma similar à apresentada na seção anterior, essa clareza na apresentação dos personagens principais permite definir e ampliar as perspectivas inicialmente colocadas nesse modelo, inserindo os personagens figurantes com o conhecimento necessário e assim facilitar a forma dos personagens figurantes de interferir na história que se desenrola.

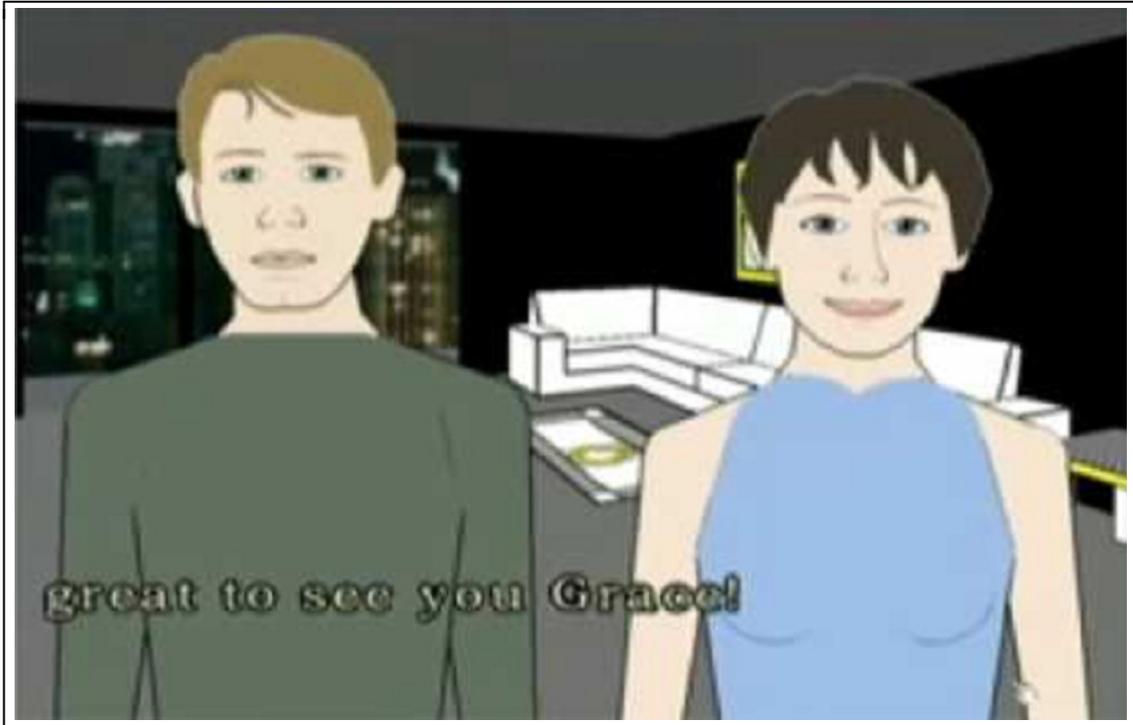


Figura 10. Screenshot do Façade.
Fonte: Mateas; Stern, 2007.

3. LOGTELL

O *Logtell* é um sistema de geração e dramatização de histórias interativas e foi utilizado para a validação do trabalho proposto nesta dissertação. O *Logtell* é uma arquitetura híbrida que compreende conceitos tanto da narrativa interativa baseada em enredos como em personagens. O objetivo do *Logtell*, que se baseia nos conceitos do Propp (1968), é permitir a interferência do usuário enquanto a história ainda mantém um rumo apropriado (Ciarlini et al., 2005). Na Figura 11, é demonstrada a arquitetura do *Logtell* com os seus módulos que serão explicados nesta sessão.

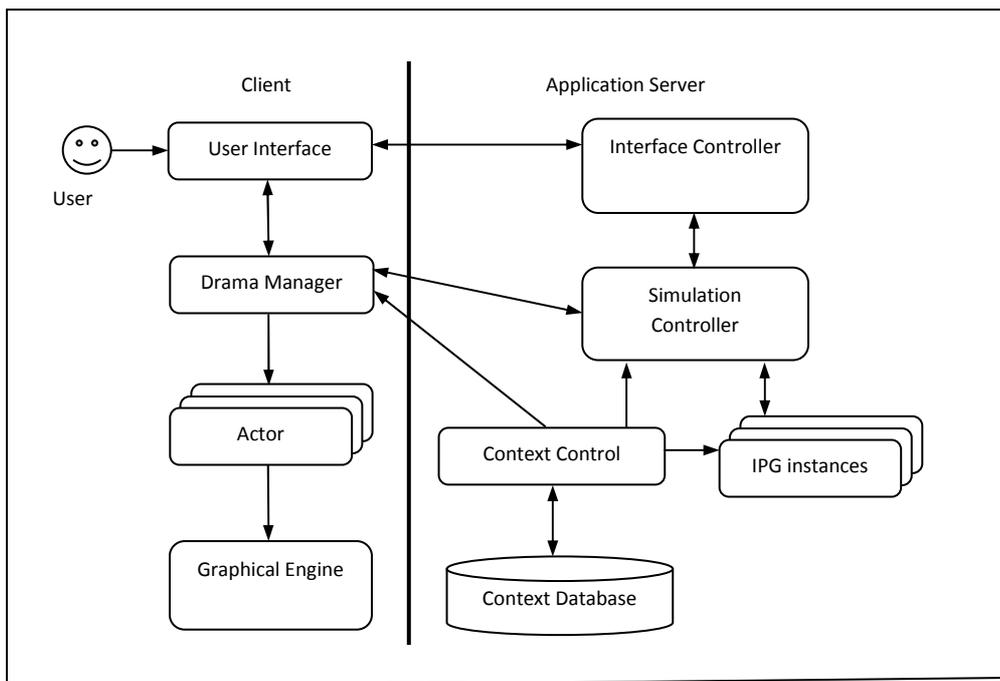


Figura 11. Arquitetura do *Logtell*.
Fonte: Camanho et al. (2009)

O sistema do *Logtell* possui uma arquitetura cliente/servidor. A estrutura servidora é quem encarrega de gerenciar os clientes e os enredos da história. O cliente é responsável pela dramatização das histórias e pela interação com os usuários. Essa configuração do sistema permite que diversos usuários possam interagir em uma ou mais histórias durante o mesmo tempo.

O módulo do *Logtell*, *Plot manager*, é o responsável pela interface do sistema com o usuário. A geração de enredos feitas pelo *Interactive Plot Manager* (IPG) é iniciada pelo *Plot Manager*, que recebe os enredos parciais gerados ao se desenvolver a estória e permite a interferência do usuário no processo de geração da estória.

O IPG é responsável por fazer simulações dos contextos especificados pelo usuário. E por fim, o *Drama Manager* é o responsável pela dramatização do enredo. Um exemplo da interface do *Plot Manager* é apresentado na Figura 12.

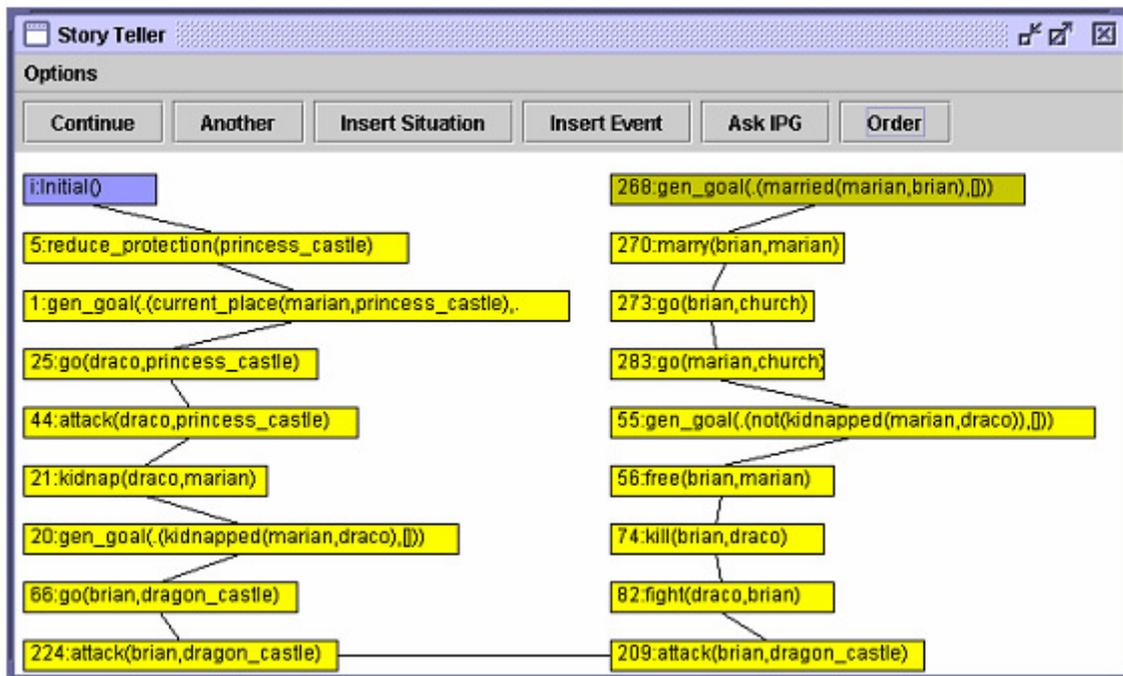


Figura 12. Interface do *Plot Manager* com um exemplo de estória gerada.
Fonte: Ciarlini et al., 2005.

O IPG gera enredos como em uma sequência de capítulos. Cada capítulo corresponde a um ciclo no qual a interferência do usuário é incorporada, objetivos são inferidos e o planejamento é usado para alcançar os objetivos (CIARLINI et al., 2008).

O IPG é um módulo de geração de enredos, implementado com o uso de técnicas de inteligência artificial e banco de dados por meio da linguagem Prolog. Ele é composto por dois submódulos: um de inferência de objetivos e outro de planejamento.

No módulo de planejamento, o IPG faz a ligação entre os eventos de acordo com a lógica, explicitada por suas pré e pós-condições. O módulo planejador do IPG é hierárquico, pois permite estabelecimento das pré-condições mais importantes, e é não linear, ou seja, estabelece relações de ordem entre eventos, apenas quando necessário o que em geral leva a uma maior eficiência (POZZER, 2005).

O IPG tem uma série de operações utilizadas para gerar histórias. Embora tais operações sirvam para compor uma história de forma interativa e coerente, elas funcionam em alto nível e existe uma carência para deixar as histórias mais enriquecidas e atraentes para o usuário (espectador). É importante destacar que, caso os filmes fossem compostos apenas por operações principais, estes não teriam o tempo de duração que possuem atualmente, caso não tivessem inseridos neles os eventos secundários.

Uma lista das operações do IPG, utilizadas na geração dos enredos com suas especificações, é apresentada a seguir.

- *Go* (CH,PL): O Personagem CH se desloca para o local PL dentro do cenário 3D.
- *Reduce_protection* (VIC, PL): Local PL de uma possível vítima VIC tem sua proteção reduzida com o número de guardas menor.
- *Attack* (CH, PL): Personagem CH ataca o local PL. Para que o ataque ocorra, o personagem CH deve ter uma natureza (boa ou má) diferente do local PL;
- *Kidnap* (VIL, VIC): Vilão VIL rapta a vítima VIC. Para acontecer o rapto, a força do vilão deve ser maior que a soma da força da vítima com a força do local onde ela se encontra;
- *Fight* (CH1, CH2): o personagem CH1 luta contra o personagem CH2. Para a luta ocorrer, os personagens devem ser de naturezas diferentes;
- *Kill* (CH1, CH2): o personagem CH2 é morto pelo personagem CH1. Assim como no evento “*Fight*”, os personagens devem ser de naturezas diferentes;
- *Free* (HERO, VIC): o herói HERO libera a vítima VIC. A vítima aumenta a afeição pelo herói;
- *Marry* (CH1, CH2): Os personagens CH1 e CH2 se casam. Para o casamento ocorrer o vilão deve ter sido derrotado e a vítima liberada;
- *Donate* (CH1, CH2): o personagem CH2 recebe força por poderes mágicos do personagem CH1.
- *Bewitch* (CH1, CH2): o personagem CH2 é enfeitiçado pelo CH1, deixando o personagem CH1 mais malvado e forte.

No mundo virtual em sua configuração inicial, a vítima é a princesa Marian, os heróis são os cavaleiros Brian e Hoel, o vilão é o dragão Draco, e o Turjan é o mago. Os locais são o *White Palace*, castelo onde mora a princesa Marian; *Green Castle*, onde habita o mago Turjan; e o *Red Castle*, onde se localiza o vilão Draco. Cada personagem tem um nível de

força, assim como cada local possui guardas protegendo. Um exemplo de sequenciamento de estória gerada pelo IPG está na Figura 13.

```

Reduce_Protection(White_Palace)
Go(Draco, White_Palace)
Attack(Draco, White_Palace)
Kidnap(Draco, Marian)
Go(Brian, Red_Castle)
Attack(Brian, Red_Castle)
Fight(Brian, Draco)
Kill(Brian, Draco)

```

Figura 13. Exemplo de estória gerada pelo IPG.

Atualmente o novo sistema de dramatização do *Logtell* está sendo feito na Unity3D (UNITY3D, 2010). *Unity3D* é uma *engine* multiplataforma para jogos. A Figura 14 apresenta uma comparação de uma estória dramatizada na versão antiga do LOGTELL (na esquerda) e a atual com a utilização da *Unity3D* (na direita). A Figura 15 apresenta os terrenos (cenários) criados na *Unity3D* para o sistema atual do LOGTELL (DALLA FAVERA, 2009).

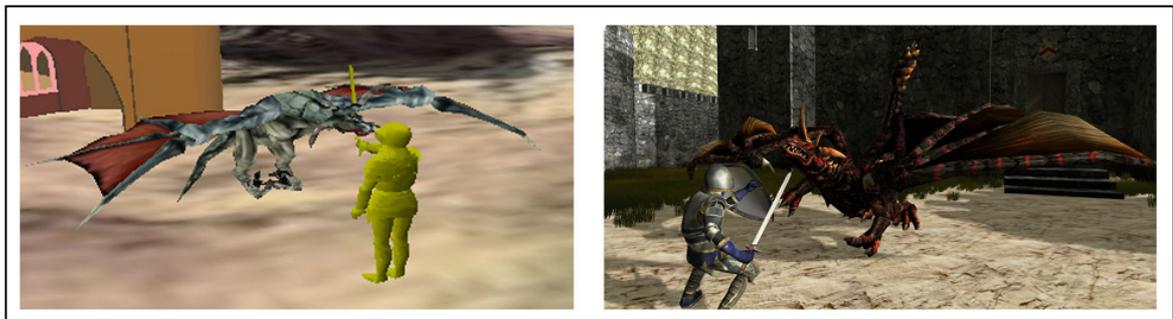


Figura 14. Comparação entre o evento “*Fight*” antigo e o atual com a *Unity3D*.
Fonte: Dalla Favera, 2009.

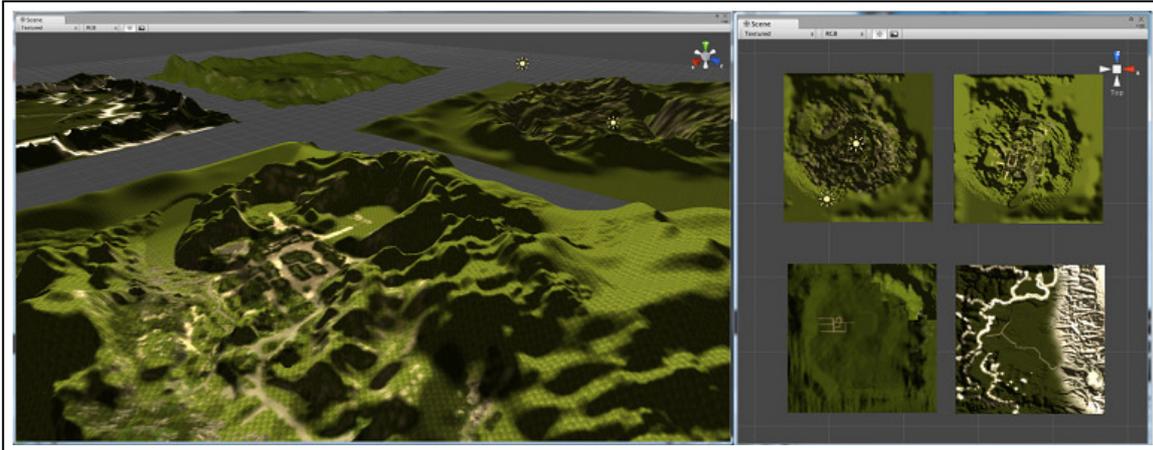


Figura 15. Terrenos criados na *Unity3D* pelo *Logtell*.
Fonte: Dalla Favera, 2009.

O diretor de figurantes estará muito ligado ao IPG, pois seu principal objetivo é aumentar o realismo e a dramatização dos eventos gerados pelo IPG, em especial dos eventos do tipo “Go”, que atualmente são bastante monótonos.

O sistema *Logtell*, como mostrado neste trabalho, se apresentou como um sistema importante e alinhado para o desenvolvimento dos objetivos desta dissertação. Esse sistema possui uma estrutura compatível e facilitada para inserir as ações necessárias para o diretor de personagens figurantes bem como das ações complementares como a melhoria e ampliação das câmeras necessárias a dar um maior realismo e dramaticidade, como o objetivo buscado no desenvolvimento desta dissertação.

4. DIRETOR DE FIGURANTES

Neste capítulo, são descritas as especificações do diretor de figurantes, a principal contribuição desta dissertação. Atualmente no *Logtel*, os eventos que carecem de maior enriquecimento dramático são o “*Go*” e o “*Fight*”, e é justamente nesses eventos que se direcionam as principais ações do diretor de figurantes com a inserção dos personagens na estória. Para exemplificar esse contexto, quando um personagem se desloca para um determinado local, a câmera mostra normalmente esse personagem andando em uma cidade vazia e monótona, ou também quando dois personagens lutam no meio de uma cidade que não tem ninguém (plateia) por volta dos personagens para acompanhar a luta.

O diretor de personagens figurantes, nesse contexto, tem como intuito enriquecer a dramatização do *Logtell*, criando e inserindo personagens figurantes que servem tanto para preencher as cidades como para interagir com personagens principais em determinados eventos gerados pelo IPG.

Para a geração dos personagens figurantes, utilizou-se a proposta de Pettre et al. (2006) vista na revisão bibliográfica, com diversas adaptações voltadas para as estórias interativas. Enquanto Pettre et al. (2006) sugere a geração de multidões bastante populosas, com o enfoque no seu deslocamento, nesta dissertação, o objetivo é gerar personagens figurantes, em um número bem mais reduzido, e com maior interação com o ambiente e com os personagens principais. A geração dos personagens figurantes é composta por dois processos, um que ocorre fora do tempo de execução da estória e outro que ocorre durante a execução da estória. Os dois processos são apresentados a seguir.

4.1. Geração de Waypoints

A geração de *Waypoints* ocorre antes do tempo de execução e apenas uma única vez para cada cenário. Essa fase consiste em gerar pontos nos quais os personagens figurantes poderão se deslocar. O modelo baseado na proposta de Pettre et al. (2006) foi escolhido pela capacidade de gerar *Waypoints* de forma dinâmica, independentemente do cenário. Dessa forma, pode-se aplicar esse modelo de geração para cada cenário do *Logtell* com pouquíssimas adaptações. Os passos para a geração de *Waypoints* são apresentados a seguir:

- **Matriz de colisão:** Considerando cada cenário do *Logtell* como uma matriz bidimensional, em que o comprimento desta é dado pela componente x do cenário, e a altura pelo componente z . Para cada cidade, aplica-se um *Raycast* para cada ponto da matriz bidimensional do cenário, verificando os pontos de possível colisão como, por exemplo, paredes, água, etc. Uma matriz é gerada armazenando valores binários indicando se cada ponto é de possível colisão (obstáculo) ou não.
- **Clearance Map:** *Clearance* é a distância de cada ponto para o ponto de borda mais próximo. Um ponto de borda é vizinho de um ponto de colisão. Para gerar o *Clearance Map*, é necessário descobrir todos os pontos de borda na matriz de colisão e posteriormente calcular o *clearance* de cada ponto. O *Clearance Map* final será uma matriz resultante na qual cada ponto indicará sua distância para o ponto de borda mais próximo.
- **Geração de Waypoints:** Para cada ponto do *Clearance Map*, são criadas circunferências cujo raio é igual ao *clearance*. Circunferências que estão completamente inseridas dentro de outra com um raio maior são ignoradas. Com todas as circunferências geradas, faz-se uma intersecção entre estas. Nos pontos em comum das circunferências, são criados corredores, e, no ponto médio, é gerado o *Waypoint*. Um exemplo ilustrativo pode ser visto na Figura 16, em que os pontos azuis correspondem aos pontos de intersecção das circunferências, as linhas azuis, que conectam os pontos azuis, ilustram o corredor, e os pontos vermelhos são os *Waypoints*. A região mais escura demonstra os obstáculos.

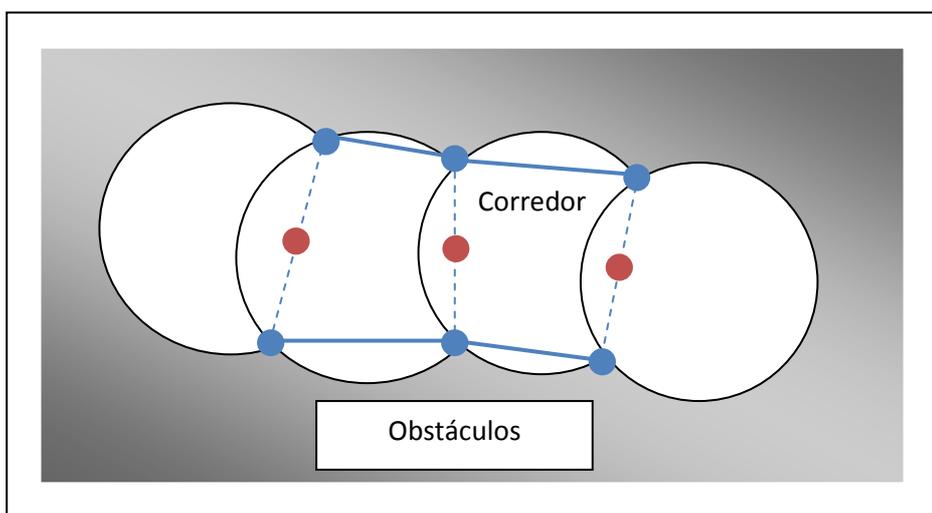


Figura 16. Ilustração da geração de *Waypoints*.
 Fonte: Adaptado de Pettre, 2006.

Para este trabalho, foram gerados, em média, aproximadamente 1.200 *Waypoints* para cada região. Com o intuito de fazer uma melhor associação entre o espaço ocupado nos cenários do *Logtell* com o espaço no mundo real, uma pessoa fica confortável no mundo real ocupando aproximadamente 1m² de área. Considerando o cenário do *Logtell* como uma matriz bidimensional em que a altura da matriz é dada pelo eixo *x* do cenário e a largura da matriz dada pelo eixo *z* do cenário, percebe-se que o espaço ocupado por um personagem no cenário apresentado foi de 4x4 da escala da *Unity3D*. Então a relação entre o espaço no cenário, em uma comparação com o mundo real, foi dada em uma relação de 4x1.

Cada cenário, em média, ocupa um espaço de 300x200 na escala da *Unity3D*. Dessa forma, cada região do *Logtell* tem um tamanho aproximado de 75mx50m, na relação de 4x1, ou seja, existe aproximadamente um *Waypoint* a cada 12,5m² (considerando a geração de 1.200 *Waypoints* por cenário). Embora esse processo de geração de *Waypoints* seja custoso, deve ser considerado que tal processamento é feito uma única vez para cada cenário (região), pois os *Waypoints* ficam armazenados em uma matriz, para serem reutilizados posteriormente. A geração de *Waypoints* em tempo real por esse processo é inviável pois demanda um tempo elevado, o que atrapalharia o andamento da estória.

4.2. Geração de personagens figurantes

A geração de personagens ocorre quando o Diretor de Figurantes recebe do *Plot Manager* um evento do tipo “Go”, que determina ao personagem principal se deslocar de uma cidade (ou cenário) para outra. Nesse evento, o Diretor de Figurantes é acionado para gerar personagens figurantes nas cidades (cenários) para o qual o personagem principal está se movimentando.

Nos primeiros testes gerados neste trabalho para simular a movimentação de personagens figurantes, houve a formação de filas indianas, o que não apresentou resultado muito satisfatório, pela falta de realismo. Uma solução encontrada para esse problema foi criar grupos de personagens figurantes que se locomovem em conjunto, permitindo assim a visualização de locomoções mais realistas, similares às que ocorrem em ambientes urbanos reais. A Figura 17 ilustra um exemplo de grupo se locomovendo.

Cada grupo possui de um a três personagens que possuem os mesmos *paths*. O path para cada grupo é escolhido de forma aleatória dentre quatro direções: norte, sul, leste ou oeste. Cada personagem figurante, que integra um determinado grupo, irá se deslocar a uma

distância máxima para a direção definida para o seu *path* até que outra direção seja determinada.

O método escolhido para os *paths* dos personagens satisfaz as necessidades requeridas para o realismo da dramatização, não sendo necessário um método mais elaborado para a definição da rota de deslocamento dos personagens. Calcular *paths* para cada personagem figurante durante o tempo de execução é muito demorado, e como cada Personagem Figurante será filmado apenas em espaços reduzidos de tempo na dramatização da estória, não se faz necessário implementar algoritmos mais rebuscados, que exigem um maior processamento.



Figura 17. Grupo de personagens se locomovendo no cenário.

Os Personagens Figurantes, quando inseridos na estória, possuem os parâmetros listados a seguir:

- **Destino:** Para qual *Waypoint* o Personagem Figurante está se locomovendo.
- **Rota:** Define o *path* no qual o Personagem Figurante se movimenta.
- **Grupo:** Define a qual grupo o Personagem Figurante pertence.
- **Velocidade:** Define a velocidade de deslocamento do Personagem Figurante.

É importante destacar que, conforme foi apresentado anteriormente, cada *Waypoint* fica num corredor pré-definido dentro do cenário. Diferentemente do trabalho de Pettre et al. (2006), que foca em simulação de multidões, o Diretor de Personagens Figurantes trata com histórias interativas que possuem personagens de diferente relevância e função dentro do contexto.

Um Evento “Go”, por exemplo, possui um Personagem Principal se deslocando e, no caminho deste, existem os Personagens Figurantes. Para o bom andamento da história, a caminhada do Personagem Principal não pode ser afetada por possíveis colisões com os personagens figurantes. Dessa forma, algumas adaptações foram feitas para tratar tal problema como na criação das rotas de deslocamento. Dentro dos corredores, estão disponíveis sete rotas, que são caminhos opcionais paralelos aos *Waypoints* principais. Cada Personagem Figurante fica em uma rota. As rotas auxiliam os personagens figurantes a andar em grupos e assim evitar um número excessivo de colisões.

A Figura 18 apresenta uma ilustração das rotas na cidade, cada uma representada por uma cor diferente. No total foram definidas sete rotas para o deslocamento, sendo a rota do meio reservada preferencialmente para o deslocamento do ator principal, e as três rotas em cada lado da rota principal disponível aos PFs, pois se definiu que os grupos podem ter de um a três personagens figurantes.

O Diretor de Figurantes não controla o *path* do Personagem Principal, mas o Personagem Principal se desloca na sua grande maioria das vezes pelo meio do corredor.

Na Figura 17, percebe-se que o meio do corredor está vazio, pois esta é a rota reservada preferencialmente para o personagem principal. Embora a rota intermediária seja reservada para personagens principais, não significa que Personagens Figurantes nunca irão utilizá-la. Dependendo da análise do tratamento de colisão dos Personagens Figurantes, esses personagens podem escolher utilizar tal rota como será explicado a seguir nesta sessão.

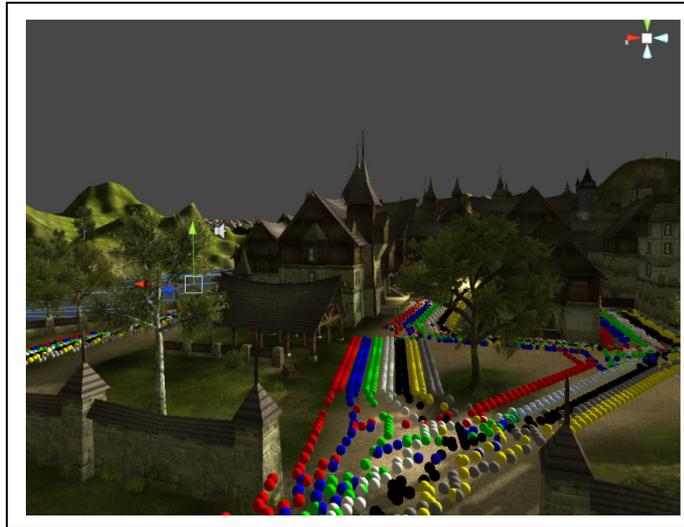


Figura 18. Ilustração das rotas no cenário.

Para tratar colisões entre personagens, cada personagem figurante possui uma *BoundingBox* e dois *colliders*: um *BoxCollider* e um *Sphere Collider*. A Figura 19 demonstra o modelo de personagens com seus *colliders*. O *SphereCollider* tem um raio maior e serve para detectar colisões com os personagens principais, enquanto o *BoxCollider* é usado para tratar colisões com os Personagens Figurantes. A *BoundingBox* serve como último recurso para evitar que os personagens figurantes colidam com os principais, o que dá a impressão que um personagem está “dentro” do outro. As regras básicas para o tratamento de colisões são tratadas da seguinte forma:

- **Colisão com Personagens Principais:** Quando o Personagem Figurante detecta colisão com o Personagem Principal ele verifica primeiramente qual a natureza do Personagem Principal. No caso da colisão detectada ser de um herói, e se o herói estiver em movimento, o Personagem Figurante pára e espera o Personagem Principal sair do seu campo de visão. Caso seja um vilão, o Personagem Figurante verifica a distância entre o vilão e o Herói. Se essa distância for pequena, provavelmente acontecerá um evento do tipo “*Fight*”. Assim o Personagem Figurante assume uma postura para torcer, preparando-se para um posicionamento para formar a plateia da luta. Se a distância entre o vilão e o herói for grande, o Personagem Figurante irá fugir.

- **Colisão com Personagens Figurantes:** Se o Personagem Figurante está vindo em direção contrária e está na mesma rota, troca de rota. A nova rota é preferencialmente a rota ao lado mais distante da rota intermediária reservada para o herói. Por exemplo, se o personagem figurante se encontra na rota 6, a rota reservada para o herói é a rota intermediária de número 4, então a primeira opção de escolha para a rota será a rota 7. Mas caso for detectada uma colisão na rota 7, é escolhida a rota mais próxima que não houver previsão de colisão, podendo ser inclusive a rota 4. Caso não esteja em direção contrária, ou seja, em direção perpendicular, o primeiro Personagem Figurante a detectar a colisão espera o outro passar.

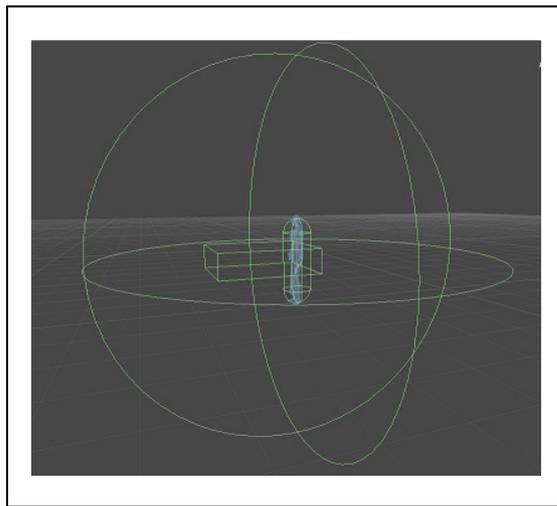


Figura 19. Modelo do Personagem Figurante com seus *colliders*.

Com os testes realizados, percebeu-se uma limitação da *Unity3D*, a *engine* utilizada para a implementação do *Logtell*. Os *TriggerEvents* dos *Colliders* apenas detectam colisão com personagens que estão em movimento. Dessa forma, quando um Personagem Principal estava parado, o Personagem Figurante não conseguia detectar essa colisão. Para resolver o problema, o *SphereCollider*, utilizado para detectar a colisão com os Personagens Principais, foi removido dos Personagens Figurantes e foi adicionado um *BoxCollider* nos Personagens Principais. Também para melhorar o desempenho, foi adicionado mais um pequeno *BoxCollider*, localizado na frente do Personagem Principal, com o objetivo dos Personagens Figurantes, que colidam primeiro com o *BoxCollider* menor, em vez de esperar o Personagem Principal passar. Isso permite ao personagem figurante desviar do Personagem Principal e não atrapalhar o caminho do Personagem principal.

Na Figura 20, é apresentado o modelo final do *colliders* para um personagem figurante específico e para um personagem principal.

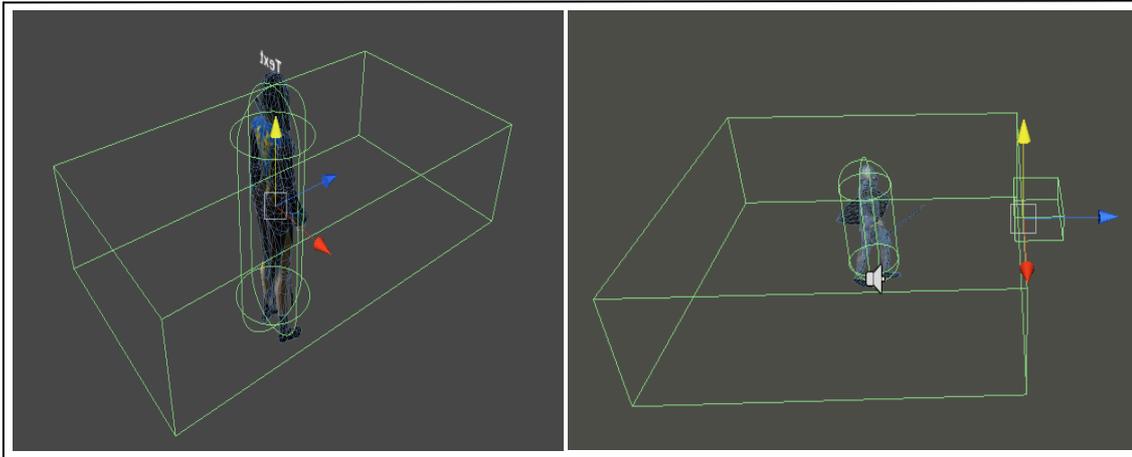


Figura 20. Modelo final do Personagem Figurante e do Personagem Principal com seus *colliders*.

4.3. Personagens Figurantes Estáticos

Nos cenários, encontram-se diversos Personagens Figurantes que não possuem os parâmetros de movimentação apresentados na sessão anterior. São personagens que se localizam em alguns pontos estratégicos e servem para ilustrar melhor as regiões, possuem animações específicas e não se locomovem. Estes são os chamados de Personagens Figurantes estáticos e podem estar na varanda de uma casa conversando ou simplesmente encostados na parede observando e povoando o cenário.

Os Personagens Figurantes estáticos foram distribuídos em pontos estratégicos pelos quais os Personagens Principais passam com maior frequência, aumentando as chances desses personagens serem filmados ao se desenvolver a história. Por exemplo, em um evento “Go” do Personagem Principal Brian, este se desloca em direção ao castelo do Dragão que é um evento comum nas histórias no *Logtell*. Nas simulações realizadas, foram inseridos em média dez personagens figurantes estáticos estrategicamente posicionados em cada região do *Logtell*, que aparecem como expectadores desses deslocamentos. Uma exceção a essa regra é na região do castelo do Dragão que não é habitada.

A Figura 21 apresenta exemplos de personagens estáticos posicionados no cenário. Na primeira parte, observam-se dois personagens estáticos conversando em uma sacada e, na segunda, um personagem observando o cenário.

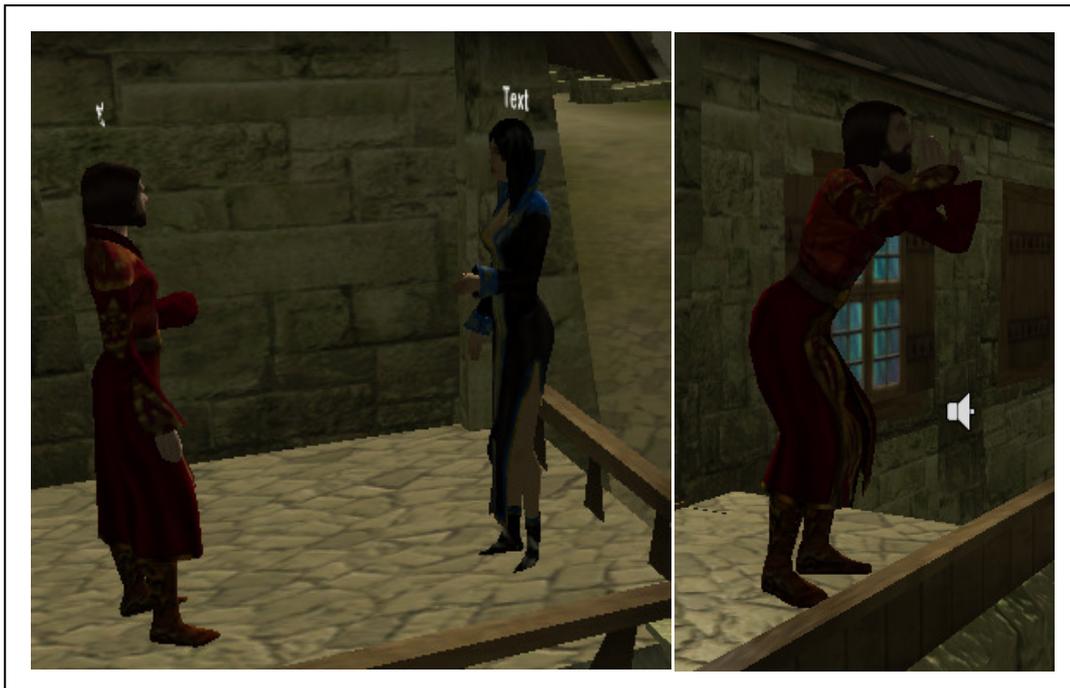


Figura 21. Personagens Figurantes Estáticos nos cenários.

4.4. Personagens Figurantes de segundo nível

Mesmo com a geração dos Personagens Figurantes já citados anteriormente, percebeu-se uma necessidade de maior interação entre os Personagens Figurantes e os Personagens Principais. Os Personagens Figurantes de segundo nível possuem essa capacidade de interação. Eles se localizam em pontos estratégicos em algumas regiões e dialogam com o personagem principal quando este passa pelo local.

Os Personagens Figurantes de segundo nível possuem um *SphereCollider*, como demonstra a Figura 22. Esse *SphereCollider* serve para detectar a colisão (ou aproximação) com os personagens principais. Quando um Personagem Principal colide com o *SphereCollider*, o Diretor de Figurantes manda uma mensagem para o Personagem Principal esperar. O Personagem Figurante irá se deslocar até próximo ao Personagem Principal e irá começar um diálogo.

Quando o diálogo terminar, o Diretor de Figurantes manda uma mensagem liberando o Personagem Principal para seguir sua rota e o Personagem Figurante de segundo nível volta a seu estado inicial. O diálogo entre o personagem principal e o personagem figurante de segundo nível dura 10 segundos e vai variar de acordo com a emoção dos personagens figurantes, que será explicada na sessão 4.8. Futuramente esses personagens figurantes de

segundo nível podem possuir papéis mais importantes nas estórias, podendo ser, por exemplo, um vendedor que irá conseguir um item para ajudar o personagem principal a matar o vilão.

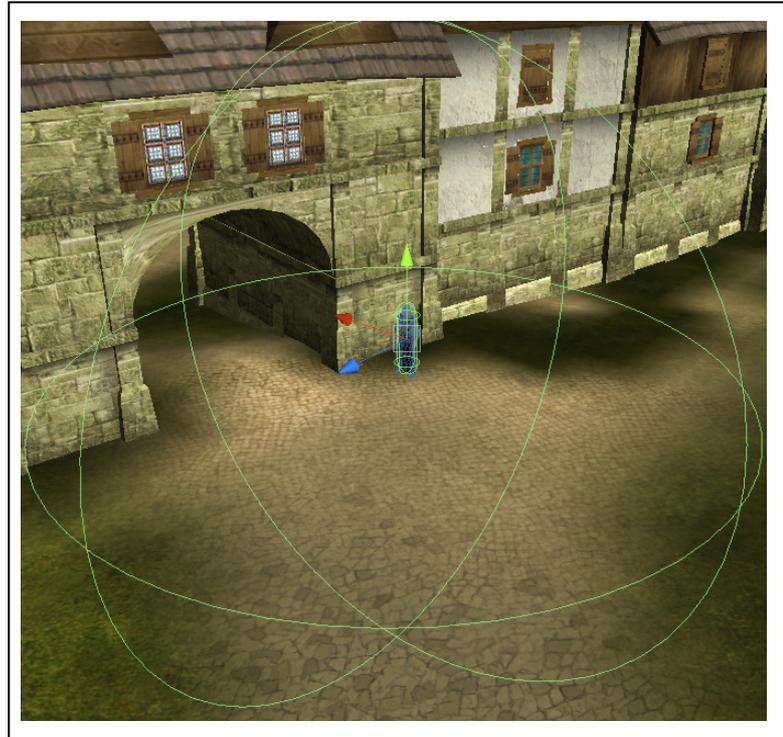


Figura 22. Personagens Figurantes de segundo nível.

4.5. Fight Scene

O Diretor de Figurantes desenvolvido neste trabalho é acionado em dois eventos do IPG: o evento “Go” e o evento “Fight”. No evento “Go”, como explicado na seção 4.2, ocorre a geração dos personagens que irão se locomover dentro dos cenários. No evento “Fight(CH1,CH2)”, o Diretor de Figurantes recebe como argumento os dois personagens principais que irão lutar.

O Diretor de Figurantes verifica quais as informações dos personagens principais e seleciona os personagens figurantes que estejam próximos ao local do evento, para formar a plateia da luta. Na Figura 23, está representado como é a formação da plateia num evento do tipo “Fight(CH1, CH2)”. O Diretor de Figurantes calcula o ponto médio entre os dois personagens principais CH1 e CH2, representado pelo ponto vermelho PM na figura. Com o ponto médio definido, é gerada uma circunferência partindo do ponto médio cujo raio é o dobro da distância do ponto médio até os personagens principais.

Na Figura 23, a distância do ponto médio para os personagens principais é representada pela linha azul, e o raio é representado pela linha vermelha. Os personagens figurantes ficam localizados na linha da circunferência. Foi escolhida essa distância para o raio, pois é uma distância confortável para os personagens figurantes se localizarem sem atrapalhar a luta e ainda assim ficarem próximos para que pudessem ser filmados.

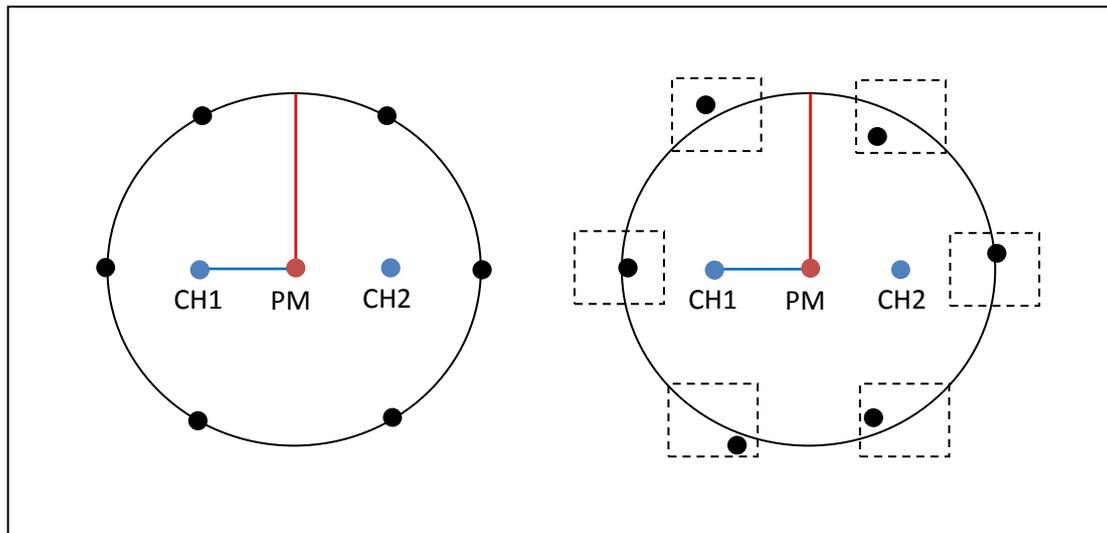


Figura 23. Formação da plateia na luta.

Nos primeiros testes da formação da plateia, foi observada a forma uniforme com o que os personagens figurantes se organizavam, dando um aspecto pouco realista. Um método encontrado para resolver este problema foi adicionar um pequeno deslocamento randômico na posição nos eixos x e z em cada personagem figurante da plateia.

Esse deslocamento produziu um resultado mais satisfatório e realista ao cenário da plateia que acompanha a luta, pois as distribuições dos personagens não ficam uniformemente posicionados. No círculo da direita da Figura 23 está representado o deslocamento aleatório dos personagens figurantes na plateia. Na Figura 24 está a comparação da plateia gerada sem o fator aleatório e depois da aleatoriedade implementada.



Figura 24. Animação dos figurantes durante a luta.

Os Personagens Figurantes que estão na plateia têm diferentes reações ao terminar a luta, dependendo do resultado. Após o evento “*Fight*”, os personagens figurantes aguardam qual o novo evento recebido pelo Diretor de Figurantes. O evento principal que ocorre após o evento “*Fight*” é o evento “*Kill*”. O Evento “*Kill*” tem como parâmetros CH1 e CH2, que são dois personagens principais que estavam em luta na qual o personagem CH1 mata o personagem CH2. Caso o vencedor da luta for o herói, os personagens figurantes vão comemorar a vitória. Já no caso de vitória do vilão, os personagens figurantes irão fugir do local da luta.



Figura 25. Diferentes reações dos figurantes durante a luta.

4.6. Manipulação de câmeras

Para uma melhor inserção dos Personagens Figurantes nas histórias, verificou-se a necessidade de câmeras específicas capazes de filmar apropriadamente a atuação deles nos

cenários. O usuário/expectador observa a atuação dos personagens figurantes com base nas filmagens realizadas pelas câmeras.

No evento “Go”, por exemplo, existiam duas câmeras no sistema do *Logtell* para filmar a caminhada do personagem principal. As duas câmeras são colocadas na mesma altura do personagem principal, alternando apenas se o personagem é filmado de frente ou de trás durante o seu deslocamento. No *Logtell*, existe o módulo *Cameraman* que é responsável por gerenciar as câmeras e verificar se existe colisão da câmera com algum objeto no cenário. Com os Personagens Figurantes no cenário, muitas vezes, tanto a câmera da frente quanto a de trás, se envolviam em colisão.

Para resolver esse problema, e também para uma melhor visualização do cenário com os figurantes, foi gerada uma nova câmera para o caso do evento “Go”. Na Figura 26, é mostrada uma cena da caminhada do personagem principal com a nova câmera.



Figura 26. Nova câmera no Evento “Go”.

Ainda para o evento “Go”, foram geradas novas câmeras para acompanhar o personagem principal ao trocar de cenário. Essas câmeras têm como objetivo apresentar para o espectador a nova região na qual o personagem principal está entrando, mostrando alguns pontos estratégicos da região ou uma visão geral de cima da região mostrando a locomoção dos Figurantes. Assim, existem quatro câmeras de apresentação para cada região.

Para evitar repetição da sequência de câmeras mostradas em diferentes regiões, apenas a região destino do personagem principal será apresentada pelas câmeras de introdução à região, independente se outras regiões intermediárias estiverem presentes no caminho designado para um personagem.

Também para evitar mostrar sempre as mesmas câmeras (posições do cenário) em cada região, apenas duas câmeras escolhidas aleatoriamente entre as quatro existentes são mostradas. Cada câmera filma um local de apresentação da região por 5 segundos, aumentando o tempo de duração do evento em 10 segundos.



Figura 27. Diferentes tomadas de câmera apresentando a região.

Para melhorar o registro de um evento “*Fight*” também foram geradas novas câmeras. Assim como no evento “*Go*”, as câmeras que já existiam no *Logtell* não conseguiam mostrar a plateia de uma forma satisfatória. Para o evento “*Fight*” foram geradas duas novas câmeras, sendo uma para capturar a plateia durante a luta, e outra para melhor visualizar a fuga dos personagens figurantes no caso de vitória do vilão. Na Figura 25, estão apresentadas as imagens geradas com as novas câmeras adicionadas para registrar o evento “*Fight*”.

4.7. Emoção dos Figurantes

Uma história gerada pelo IPG contém apenas eventos principais, como apresentado na Figura 13. Na história default, o vilão Draco ataca o *White Castle* e rapta a princesa. O herói Brian vai até o *Red Castle* e luta contra o vilão Draco. Brian mata Draco e salva a princesa. Marian e Brian se casam. Essa história contém apenas eventos principais com poucos

parâmetros. O sistema permite interação do usuário, que pode desejar, por exemplo, que o vilão seja o vencedor da luta, alterando assim o rumo da história. Dessa forma, o Diretor de Figurantes não sabe o que acontecerá nos próximos eventos, reduzindo assim a capacidade de expressividade dos personagens.

Um método encontrado para aumentar a expressividade dos Personagens Figurantes na história foi adicionar um novo parâmetro nos eventos recebidos pelo Diretor de Figurantes que expressa a emoção da cena. Como esse parâmetro não está incorporado ao planejador (IPG), ele foi adicionado “manualmente” com propósito de validação da técnica proposta com o objetivo de avaliar o resultado da sua inserção para aumentar a dramaticidade e o realismo da cena.

Em vez do Diretor de Figurantes receber o comando “Go(CH1,PL)”, ele recebe o “Go(CH1,PL,EM)” no qual EM é o parâmetro que indica a emoção que os personagens figurantes devem expressar durante a cena.

Deve-se ressaltar que cabe ao planejador fornecer essa informação, dado o fato que durante o planejamento da história este sabe se, por exemplo, um evento do tipo “Go(CH1,PL)” representa um simples passeio ou a tensa jornada para libertação da princesa que está sob as garras do malvado dragão. Com essa informação, os personagens figurantes podem adaptar seu comportamento a cada tipo de situação, aumentando assim o realismo e o aspecto dramático da história.

Os quatro tipos de parâmetros da emoção para os personagens de Figurantes são apresentados na Figura 28 e detalhados a seguir:

- **Terror:** Os Personagens Figurantes têm medo do Personagem Principal. Pode ser usado quando um vilão está indo realizar uma ação maldosa;
- **Success:** Os Personagens Figurantes celebram o sucesso do Personagem Principal;
- **Quest:** Os Personagens Figurantes incentivam o Personagem Principal na sua missão; e
- **Normal:** Personagens Figurantes não têm maiores reações em relação ao Personagem Principal.



Figura 28. Diferentes tomadas de câmera com as emoções.

O parâmetro de Emoção dos Figurantes serve ainda para os Personagens Figurantes de segundo nível. Dependendo do parâmetro de emoção recebida pelo Diretor de Figurantes, o diálogo e a postura entre o Personagem Figurante de segundo nível e o Personagem Principal será influenciado. Na Figura 29, está o dialogo entre o personagem figurante de segundo nível e o personagem principal quando a emoção passada por parâmetro for “*Quest*”. Para cada emoção, existe um diálogo pré-definido, gerado pelo Diretor de Figurantes. Para este trabalho o número de diálogos é limitado a um para cada parâmetro de emoção, e tem como propósito apenas a verificação do funcionamento da ideia proposta.



Figura 29. Diálogo do Personagem Figurante de segundo nível e Personagem Principal.

4.8. Arquitetura com *Logtell*

Nesta seção, é detalhado como o diretor de figurantes foi inserido na arquitetura do *Logtell* e o relacionamento com os outros gestores do ambiente (Figura 30).

O Roteirista faz a comunicação entre o IPG e o sistema de Dramatização. A função do Roteirista é receber o evento do *Logtell* e detalhar esse evento em diversos subeventos para repassar essa sequência para o Diretor-Geral. O Diretor-Geral, no *Logtell*, é o responsável por gerenciar e fazer a ligação com os demais diretores e o editor, além de enviar comandos para os atores. O Diretor recebe as informações de eventos do IPG enviado pelo Roteirista.

Dentro da estrutura tem também o Diretor de Fotografia, que é o responsável pelo aspecto visual do cenário e o Diretor de Música, que tem a função de trabalhar com as trilhas sonoras da dramatização. Também faz parte da estrutura o *Cameraman*, que cria um conjunto de possíveis tomadas para filmar uma determinada cena e enviar para o Editor. O Editor, por sua vez, tem como objetivo selecionar a melhor cena para a filmagem (LIMA et al., 2009).

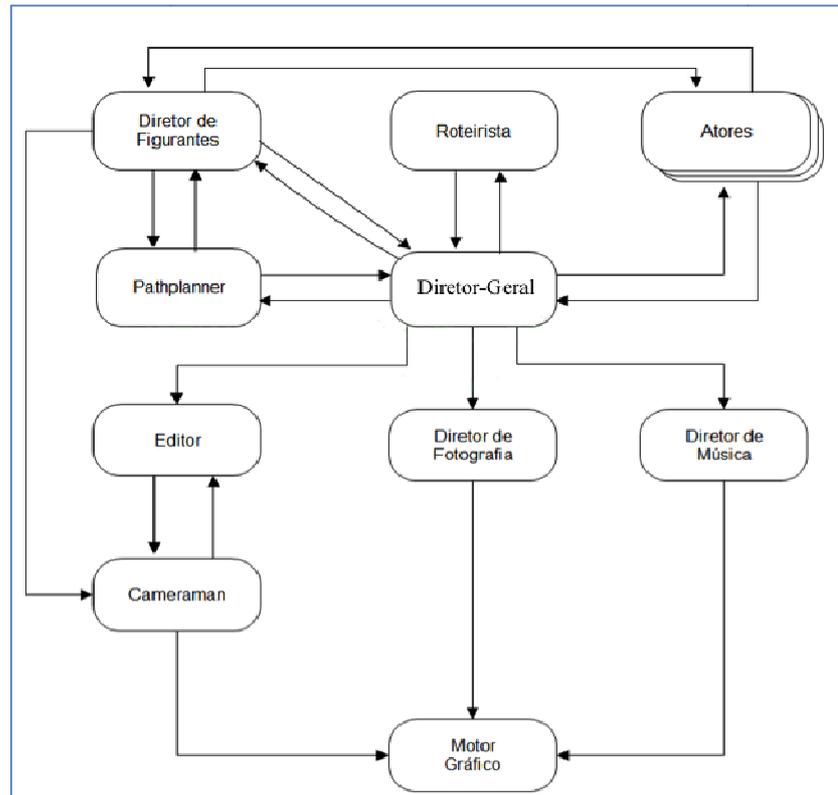


Figura 30. Arquitetura do sistema Logtell.

O Diretor de Figurantes também é acionado pelo Diretor-Geral. Quando o Diretor-Geral recebe um novo evento do Roteirista ele vai acionar o Diretor de Figurantes caso esse evento seja do tipo “Go” ou “Fight”.

No caso do Evento “Go”, o Diretor-Geral aciona tanto o Diretor de Figurantes como o *Pathplanner*. O *Pathplanner* tem como objetivo determinar o *path* que o Personagem Principal, passado por parâmetro, tomará para se deslocar do local atual até o lugar de destino. Esse *path* gerado pelo *Pathplanner* varia de acordo com o interesse e a ação do usuário. No caso do usuário optar por uma história com maior duração, o *path* gerado será mais detalhado, e o Personagem Principal passará por mais locais do cenário.

Ainda no caso do evento “Go”, o Diretor de Personagens Figurantes vai gerar personagens figurantes na região do *Path* criado pelo *Pathplanner* para dar maior realismo e aprimorar a dramaticidade das caminhadas do personagem principal. A localização dos personagens principais no cenário é importante, pois além do Diretor de Personagens

Figurantes preparar a dramatização nos locais próximos ao personagem principal, também é necessário evitar colisões, para não atrapalhar o desenvolvimento da estória principal.

No caso do evento “*Fight*”, o Diretor de Figurantes fica encarregado de gerar uma multidão para ficar de plateia da luta, sendo necessário saber a localização dos dois personagens que estão lutando. Os personagens figurantes, que estiverem próximos da localização dos personagens principais, são selecionados para formar a plateia da luta. Quando o Diretor de Figurantes recebe o evento “*Fight*”, vai ocorrer um corte de cena para mostrar o diálogo entre os personagens principais; quando a câmera for novamente posicionada para filmar a luta, os personagens figurantes vão estar posicionados para torcer.

Na maior parte da execução do Diretor de Figurantes, os outros Diretores continuam executando as suas funções pré-estabelecidas. Por exemplo, no evento “*Go*”, o Diretor de Figurantes roda em paralelo ao Diretor-Geral na maior parte do tempo, com exceção de dois momentos. O primeiro é quando o Personagem Principal troca de região (cenário). Nesse momento, com as câmeras filmando a introdução da nova região, o Diretor de Figurantes pede permissão para parar o andamento da estória principal e assim poder mostrar a região e os Figurantes para o espectador. O outro momento é no diálogo do Personagem Figurante de segundo nível, no qual o Diretor de Figurantes aciona os atores pedindo para o personagem principal esperar enquanto o Personagem Figurante se aproxima e começa a estabelecer o diálogo.

5. IMPLEMENTAÇÃO

Para os testes de implementação desta dissertação, foi utilizado a engine *Unity3D*. Ela é uma *game engine* que facilita a criação de cenários 3D. Essa *engine* é uma ferramenta com uma grande variedade de recursos em se tratando de facilidade e praticidade no manuseio. As linguagens aceitas para programação da *Unity3D* são: *JavaScript*, *Boo* e *C#*.

Para a implementação do Diretor de Figurantes nesta dissertação, foi utilizada a linguagem *C#*. O *C#* é uma linguagem de programação orientada a objetos desenvolvida pela Microsoft como parte da plataforma *NET*. A sintaxe da linguagem se assemelha com o *C++* com influências de outras linguagens de programação, como *Java*.

A *Unity3D* oferece ótima qualidade gráfica e suporta diversos tipos de modelos 3D que atendem à necessidade estabelecida para a implementação desta dissertação. A *Unity3D* ainda permite a criação de *shaders*, além de já oferecer diversos *shaders* pré-definidos. A física é tratada internamente utilizando a *engine Ageia PhysX*, que suporta um simulação realista (UNITY3D, 2011).

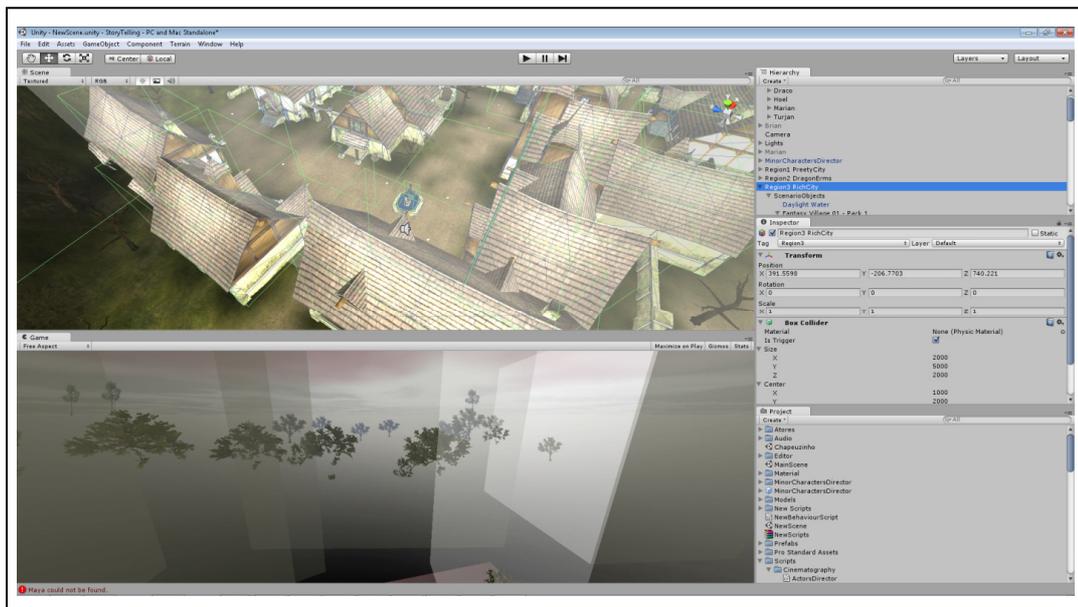


Figura 31. Interface da *Unity3D*.

Um projeto na *Unity3D* consiste em uma ou mais cenas. Cada cena consiste de *GameObjects*, *Components*, *Assets* e *Prefabs*. Um *GameObject* é um bloco de diversos

Components. Um *Component* é uma representação visível de entidades como materiais, terrenos ou partículas, ou ainda pode ser algo mais abstrato como câmeras e luzes. Um *Component* está sempre anexado a um *GameObject*.

Cada script possui duas principais funções: *OnStart()* e *OnUpdate()*. A função *OnStart* é chamada toda vez que a aplicação é inicializada, antes da primeira vez que o script rode a função *OnUpdate*. A função *OnUpdate* é chamada uma vez por *frame*. Em um *frame* composto por dez scripts, por exemplo, dentro de um projeto na *Unity* a cada ciclo da função *OnUpdate*, cada um dos dez scripts vai rodar.

Um dos principais recursos da *Unity3D* é o de *colliders*. Esse recurso permitiu a implementação do sistema de detecção de colisões utilizado neste trabalho. A utilização do *BoxCollider* e do *SphereCollider* foram muito importantes para que o sistema de detecção de colisão proposto neste trabalho funcionasse. As funções dos *TriggerEvents* dos *colliders* facilitaram a manipulação das colisões entre os personagens. Porém, um problema encontrado na *Unity* foi uma limitação de detecção dos eventos *triggers* quando objetos estão parados. Apenas objetos em movimento ativam os *TriggerEvents* dos *Colliders*. Isso foi um problema pois, muitas vezes, o personagem principal se encontrava parado e não era detectado no sistema de colisão dos personagens figurantes.

Algumas funções existentes na *Unity* para os *TriggerEvents* são: *OnTriggerEnter*, *OnTriggerExit* e *OnTriggerStay*. A função *OnTriggerEnter* é chamada quando algum objeto entra no *collider* que está anexado ao script, *OnTriggerExit* é chamado quando o objeto sai do *collider* e *OnTriggerStay* é chamado uma vez para cada *frame* em que o objeto se encontra dentro do *collider*.

Para a implementação do Diretor de Figurantes proposto nesta dissertação, diversos recursos da *Unity* foram utilizados. Para a geração dos *Waypoints* explicados na sessão 4.1, foi utilizado o recurso do *RayCast*. No código exemplo abaixo, está a função que verifica quais pontos do cenário são obstáculos. Na sua utilização, são fornecidos por parâmetros dois pontos bidimensionais, informando onde começa e termina a região nos eixos *x* e *z*. Na sequência, são executados dois laços de repetição *for* para percorrer todos os pontos da região. Em cada ponto, é feito um *RayCast* vertical de cima para baixo verificando onde houve colisão. Os parâmetros informados na função *Physics.Raycast* são: ponto 3D donde o *RayCast* é lançado, direção do *RayCast*, e a variável de colisão que vai armazenar a informação do ponto colidido. Se o ponto colidido coincidir com um possível deslocamento dos personagens figurantes, esse ponto na matriz vai receber valor positivo.

```

void testCollisions(Vector2 from, Vector2 to) {
    for(int i=(int)from.x;i<(int)to.x;i++) {
        for(int j=(int)from.y;j<(int)to.y;j++) {
            RaycastHit hit;
            Vector3 test = new Vector3(i,500,j);
            matr[i,j] = 0;
            if (Physics.Raycast (test, -Vector3.up, out hit))
                if(hit.transform.tag=="walkpoint")
                    matr[i,j] = 1;
        }
    }
}

```

Também é importante destacar que durante o processo de validação das funcionalidades do Diretor de Personagens Figurantes foram simuladas diversas estórias com diferentes níveis de dramaticidade e de viabilidade de interação do espectador. Nesse contexto, foi observado que o ambiente, para os objetivos deste trabalho, atende às necessidades junto com a *engine Unit3D*.

6. RESULTADOS

Neste capítulo, são apresentados os resultados do trabalho realizado com a incorporação do Diretor de Figurantes no ambiente de teste do *Logtell*. Para a apresentação dos resultados é mostrado nas figuras uma comparação entre a nova versão e a versão original do *Logtell* sem a interferência e atuação do Diretor de Figurantes com o ambiente. Como visto no trabalho, o Diretor de Figurantes atua sobre três principais eventos do *Logtell*: “*Go*” – que define o deslocamento dos personagens principais nos cenários; “*Fight*” – que apresenta a luta entre os personagens principais (herói e vilão); e “*Kill*” – evento que define qual o personagem que vence a luta entre o herói e o vilão.

No evento “*Go*”, é no qual o Diretor de Figurantes tem maior atuação durante a dramatização e possui maiores perspectivas de ampliar a dramatização da estória que se desenvolve. São apresentados desde personagens figurantes que se deslocam como simples cidadãos pela cidade, até personagens figurantes que interagem e dialogam diretamente com o personagem principal.

Na Figura 32, é apresentada uma comparação de imagens da versão do *Logtell* sem a utilização do Diretor de Figurantes com a versão utilizando o modelo proposto nesta dissertação. Na primeira linha, são apresentados os personagens figurantes se locomovendo durante a caminhada do personagem principal do evento “*Go*”. Pode ser observado que, com a presença dos personagens figurantes, a passagem do personagem principal é mais real e interessante para o espectador. Na segunda linha da figura é mostrada a presença dos personagens figurantes estáticos na varanda de uma casa e na terceira linha da figura, é mostrada a interação de um personagem figurante de segundo nível dialogando com o personagem principal.

Versão antiga do Logtell	Versão nova do Logtell com Diretor de Figurantes
	
<p>Personagens figurantes se locomovendo durante a caminha do personagem principal no evento "Go"</p>	
	
<p>Personagens figurantes estáticos no cenário</p>	
	
<p>Personagem figurante de segundo nível</p>	

Figura 32. Versão antiga do Logtell e a nova versão com o Diretor de Figurantes.

O evento “*Fight*” é outro evento em que atua Diretor de Figurantes. O Diretor de Figurantes é encarregado de gerar uma plateia de Personagens Figurantes que são capazes de interagir de acordo com o desenrolar da luta. Na Figura 33, é novamente feita uma comparação de imagens entre as versões do *Logtell* em cenas de luta.

Além de gerar personagens figurantes capazes de interagir com os personagens principais e enriquecer o realismo das histórias geradas, o Diretor de Figurantes manipula e gera câmeras. Essas manipulações foram necessárias para que as câmeras se adequassem aos personagens figurantes. A Figura 33 apresenta um exemplo do decorrer de uma história gerada pelo *Logtell*, com a sequência de eventos “*Go*(Draco, Gray_Castle, Terror)”, “*Fight*(Draco, Brian, Terror)” e “*Kill*(Draco, Brian, Terror)”. Na primeira linha, são apresentadas imagens da luta dos personagens principais, sendo a da direita com a plateia formada pelos personagens figurantes. Na segunda linha da figura, é apresentada a reação de comemoração dos personagens figurantes com a vitória do herói. Na terceira linha da figura, é exibida a viabilidade da câmera em capturar a imagem dos personagens figurantes se afastando do cenário da luta com a vitória do vilão.

Versão antiga do Logtell	Versão nova do Logtell com o Diretor de Figurantes
	
Personagens figurantes numa cena de luta	
	
Reação dos personagens figurantes com a vitória do herói	
	
Reação dos personagens figurantes com a vitória do vilão	

Figura 33. Comparação entre as versões do Logtell.

Na Figura 34, é apresentada uma sequência de cenas capturadas de uma estória utilizada para validar a inserção dos personagens figurantes no ambiente do *Logtell*.

- a) **Draco saindo do Red Castle:** nessa imagem, mostra o vilão “Draco” saindo do *Red Castle*.
- b) **Draco voando pelo Red Castle:** na sequência da cena, o vilão “Draco” sobrevoa os arredores do *Red Castle*.
- c) **Draco passando pelo White Castle:** no sobrevôo de “Draco” ele passa sobre o *White Castle* em que também podem ser observados personagens figurantes em suas redondezas.
- d) **Primeira câmera de apresentação do local destino, o Gray Castle:** considerando que “Draco” se aproxima do *Gray Castle*, imagens desse castelo são apresentadas pelas câmeras de introdução do personagem principal no novo cenário.
- e) **Segunda câmera de apresentação do Gray Castle:** considerando que duas das quatro câmeras criadas de apresentação exibem imagens do local (cenário) de destino, essa segunda câmera apresenta uma imagem aérea do *Gray Castle*.
- f) **Draco se locomove na Gray Castle:** O personagem principal Draco chega ao *Gray Castle*.
- g) **Draco aterrissando no Gray Castle:** o personagem principal Draco chega ao *Gray Castle* para desafiar Brian.
- h) **Draco e Brian lutando:** imagem exibindo a luta entre os personagens principais Brian e Draco. Mostra também os personagens figurantes na plateia da luta.
- i) **Reação dos personagens figurantes após Draco matar Brian:** o vilão Draco mata Brian na luta no *Gray Castle*. Os personagens figurantes fogem do local da luta.

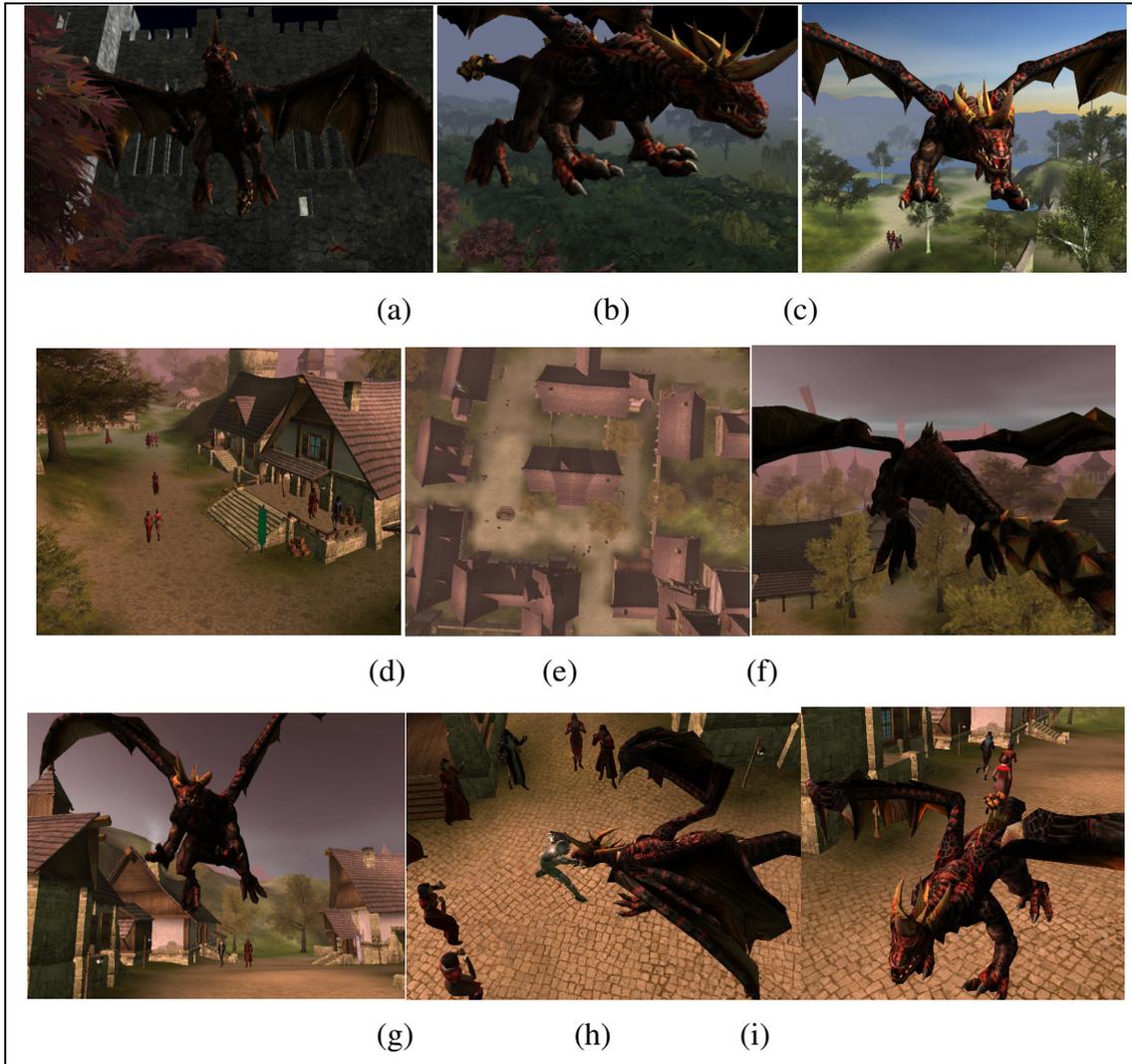


Figura 34. Exemplo de sequência de estória. (a) Draco decolando do Red Castle. (b) Draco voando pelo Red Castle. (c) Draco passando pelo White Castle. (d) Primeira câmera de apresentação do local destino, o Gray Castle. (e) Segunda câmera de apresentação do Gray Castle. (f) Draco se locomovendo no Gray Castle. (g) Draco aterrissando no Gray Castle. (h) Draco e Brian lutando. (i) Reação dos personagens figurantes após Draco matar Brian.

7. CONCLUSÃO

Nesta dissertação, foi apresentado um modelo de Diretor de Personagens figurantes, que foi idealizado, desenvolvido e incorporado ao sistema de Narrativa Interativa *Logtell*. Para o desenvolvimento das suas funções dentro das histórias, os personagens figurantes gerados têm a função de povoar o ambiente no qual se desenvolve o enredo, bem como interagir com os personagens principais da história. Com sua inserção no *Logtell*, atendeu-se aos objetivos deste trabalho que foi enriquecer a dramatização das histórias permitindo assim aumentar o tempo de duração do drama.

Para inserir os personagens figurantes nas histórias, dois problemas podem ser destacados e que exigiram uma ação especial na sua implementação. O primeiro se refere a colisões dos personagens figurantes no cenário. Para sua solução, foram definidos *paths* preferenciais para a locomoção dos personagens figurantes e definidos critérios que prevenissem as colisões. Um segundo problema ocorrido na inserção dos personagens se refere ao aspecto visual acarretado pelo fato dos personagens andarem em filas. Este foi resolvido compartilhando-se o *path* para um conjunto de figurantes, permitindo com que os personagens figurantes se locomovam em grupos. Um terceiro problema foi observado na inserção dos personagens figurantes na formação de platéia de lutas de um evento “*Fight*”. Neste, um deslocamento da posição dos personagens figurantes permitiu uma distribuição menos uniforme e com melhor efeito visual para o usuário/espectador.

Também foram geradas novas câmeras para permitir uma melhor visualização dos personagens figurantes nos cenários. As câmeras adicionais foram inseridas para melhor capturar a dramatização dos eventos “*Go*” (no qual os personagens principais se movimentam por diferentes locais interagindo com os figurantes) e “*Fight*” (no qual os personagens figurantes formam a platéia para acompanhar uma luta envolvendo os personagens principais).

Também merece ser destacada a inclusão do parâmetro de emoção. Os personagens figurantes foram capazes de ter diferentes reações em diferentes situações, dando maior realismo e emoção para as histórias. Os personagens figurantes estáticos situados nas varandas e calçadas, assim como os personagens figurantes que se locomovem pelas regiões,

enriqueceram os cenários e o realismo nas tomadas das câmeras para exibir detalhes aos espectadores.

Os personagens figurantes de Segundo nível foram capazes de aumentar o tempo de duração das histórias e também aumentar a interação entre os personagens figurantes com os personagens principais. As novas câmeras foram capazes de apresentar adequadamente os personagens figurantes, e as tomadas de câmeras que introduzem as cenas ajudaram a aumentar o tempo de duração das histórias e também dar maior realismo na dramatização. Dessa forma, foram alcançados os objetivos deste trabalho.

7.1. Trabalhos Futuros

Considerando que o trabalho realizado está inserido dentro de um contexto de histórias interativas e que objetiva-se que os usuários e espectadores tenham a perspectiva de interferir no desenvolvimento da história e visualizar um ambiente com maior dramaticidade, este trabalho inseriu, no contexto das histórias, os personagens figurantes e câmeras adicionais para exibir a participação desses personagens no drama gerado.

Diversas ampliações para conseguir melhorar os objetivos propostos nesta dissertação podem ser destacados:

- **Funções:** ampliar as funções que podem ser desempenhados pelos personagens figurantes. Nessas funções adicionais, podem ser destacados funções que tenham uma relação direta com o evento que se desenvolve. Em caso de uma luta, por exemplo, poderia existir um atendente que socorresse o herói no caso deste perder a luta. Poderiam ser desenvolvidos personagens que carreguem em uma maca o herói ferido. Outra função, que poderia ser ocupada por um personagem figurante, é o de receber os votos do casamento dos heróis da história.
- **Personagens figurantes de terceiro nível:** os personagens figurantes de terceiro nível podem ter a função de interagir entre eles mesmos, com o intuito de gerar novas cenas sem a presença dos personagens principais. Um exemplo seria quando um personagem principal se desloca de uma região para outra, em vez de mostrar direto o personagem principal na nova região como se ele tivesse sido teletransportado, mostrar uma cena de alguns personagens figurantes em um bar discutindo sobre a missão do personagem principal, para depois mostrar o

personagem principal na nova região. Com tais personagens, aumentaria ainda mais o tempo de duração das histórias.

- **Diálogos:** ampliar as viabilidades de interação e diálogos entre os personagens principais e os personagens figurantes de segundo nível, ou ainda entre os personagens figurantes de terceiro nível no caso do trabalho futuro proposto.
- **Novas câmeras:** Criar câmeras para realçar ainda mais a participação dos figurantes nas diversas ações que estes realizam.
- **Novas formas de interação:** permitir que o usuário/espectador tenha novas perspectivas de interagir com a história que se desenvolve por meio de personagens figurantes. Um exemplo seria o usuário escolher qual item o personagem principal iria comprar de um vendedor (personagem figurante) para ajudar em sua missão.
- **Funcionalidades do ambiente:** permitir que sejam inseridos parâmetros adicionais nos eventos do *Logtell*, pelo IPG, que facilitem informar a emoção a ser demonstrada pelos personagens figurantes.

BIBLIOGRAFIA

AZEREDO, V. C.; DALLA FAVERA, E. C.; POZZER, C. T. **Using Navigation Meshes to Improve Storytelling Dramatization.** In: VII Brazilian Symposium on Computer Games and Digital Entertainment, p. 21-24, november 2008.

BRAUN, U.; IURGEL, I.; GRASBON, D. Setting the scene: playing digital director in interactive storytelling and creation. Spierling. **Elsevier Journal Computers & Graphics.** vol. 26/1, Amsterdam 2002.

CAVAZZA, M.; CHARLER, F.; MEAD, S. J. Agents' Interaction in Virtual Storytelling. Intelligent Virtual Agents, **Lecture Notes in Computer Science.** Madrid, Spain. 2001.

CAVAZZA, M.; CHARLES, F. Dialogue Generation in Character-based Interactive Storytelling. **AAAI First Annual Artificial Intelligence and Interactive Digital Entertainment Conference.** Marina del Rei, Califórnia, EUA. 2005.

CAMANHO, M. M.; CIARLINI, A. E. M.; FURTADO, A. L.; POZZER, C.; FEIJÓ, B. A Model for Interactive TV Storytelling. **VIII Brazilian Symposium on Digital Games and Entertainment.** Rio de Janeiro, Brasil, 2009.

CIARLINI, A. E. M.; POZZER, C. T.; FURTADO, A. L.; FEIJÓ, B. A Logic-Based Tool for Interactive Generation and Dramatization of Stories. In: **ACM-SIGCHI International Conference on Advances in Computer Entertainment Technology**, 2005, Valencia. Proceedings, 2005.

CIARLINI, A. E. M.; CAMANHO, M. M.; DORIA, T. R.; FURTADO, A. L.; POZZER, C. T.; FEIJÓ, B. Planning and Interaction Levels for TV Storytelling. In: **First Joint International Conference on Interactive Digital Storytelling - ICIDS 2008**, Erfurt. Proc. First Joint International Conference on Interactive Digital Storytelling - ICIDS 2008. Berlin : Springer, 2008. v. 5334. p. 198-209.

DALLA FAVERA, E. C. **Técnicas de gerenciamento de cenários para aumento do tempo de dramatização em Storytelling.** Trabalho de conclusão de curso. UFSM. 2009.

GARCÍA-ROJAS, A.; GUTIÉRREZ, M.; THALMANN, D. Simulation of Individual Spontaneous Reactive Behavior. **In Proceedings of AAMAS (1)'2008.** Pp. 143-150. 2008.

- GALEF, D. **The Supporting Cast. A study of Flat and Minor Characters.** Pennsylvania State University Press. 1993.
- LIMA, E. S.; POZZER, C. T.; CERETTA, E.; CIARLINI, A.; FEIJÓ, B.; FURTADO, A.; DORNELLAS, M. Support Vector Machines for Cinematography Real-Time Camera Control in Storytelling Environments. In: **VIII Brazilian Symposium on Computer Games and Digital Entertainment**, Rio de Janeiro, p. 176-184, october 2009.
- LIMA, E.E.S. **Um modelo de dramatização baseado em agentes cinematográficos autônomos para Storytelling Interativo.** Dissertação de mestrado. UFSM. 2010.
- MASSIVE SOFTWARE. <http://www.massivesoftware.com/>. Ultima visita em Janeiro de 2009.
- MAÏM, J.; HAEGLER, S.; YERSIN, B.; MUELLER, P.; THALMANN, D.; GOOL, L. V. Populating Ancient Pompeii with Crowds of Virtual Romans. In **Proceedings of the 8th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST'07)**. 26-30. 2007.
- MATEAS, M; STERN, A; **Façade: An Experiment in Building a Fully-Realized Interactive Drama.** Game Developers Conference, San Jose, CA. 2003.
- McCRAE, R.; JOHN, O. An introduction to the five-factor model and its applications. **Journal of Personality**. vol 60/2, pages 175-215. june1992.
- MILLER, C. H. **Digital Storytelling – A creators Guide to Interactive Entertainment.** Burlington, MA: Focal Press Elsevier. 2004.
- MUSSE, S. R.; THALMANN, D. A Model of Human Crowd Behavior: Group Inter-Relationship and Collision Detection Analysis. **Proc. Workshop of Computer Animation and Simulation of Eurographics'97**. 1997. Budapest, Hungary.
- PETTRE, J.; CIECHOMSKI, P. H.; MAIM, J.; YERSIN, B.; LAUMOND, J. P.; THALMANN, D. Real-time navigating crowds: scalable simulation and rendering. **Computer Animation and Virtual Worlds**, vol. 17/3-4, Pages 445-455, Special Issue: CASA 06 (2006).
- POZZER, C.T. **Um sistema para geração, Interação e Visualização 3D de Histórias para TV Interativa.** Tese de Doutorado, PUC-Rio, 2005.
- PROPP, V. **Morphology of the Folktale.** Austin: University of Texas Press. 1968.

THALMANN, D.; MUSSE, S. **Crowd Simulation**. London : Springer. 2007.

TOBIAS, R.. **20 Master Plots and How to Build Them**. Writers Digest Books; 1st edition. 1993.

REYNOLDS, C. W. Steering Behaviors for Autonomous Characters. **Proceedings of the Game Developers Conference**, Pages 763-782, 1999.

UNITY3D, <http://www.unity3d.com/>. Ultima visita em Janeiro de 2011.