

Bruno Baère Pederassi Lomba de Araujo

**Um estudo sobre adaptatividade dinâmica de
dificuldade em jogos**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática da PUC-Rio

Orientador: Prof. Bruno Feijó

Rio de Janeiro
Setembro de 2012



Bruno Baère Pederassi Lomba de Araujo

**Um estudo sobre adaptatividade dinâmica de
dificuldade em jogos**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Bruno Feijó

Orientador

Departamento de Informática — PUC-Rio

Prof. Alberto Barbosa Raposo

Departamento de Informática — PUC-Rio

Prof. Maria das Graças de Almeida Chagas

Departamento de Artes & Design — PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 6 de Setembro de 2012

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Bruno Baère Pederassi Lomba de Araujo

Graduou-se em Engenharia de Computação na Pontifícia Universidade Católica do Rio de Janeiro em 2009 com domínio adicional em Estudos Avançados em Matemática. Desenvolve jogos independentes, tendo obtido com Lights Out o terceiro lugar no voto popular no Festival de Jogos Independentes na categoria Art Game do SBGames 2011 e sua apresentação no FILE SP 2012 – Festival Internacional de Linguagem Eletrônica.

Ficha Catalográfica

Araujo, Bruno Baère Pederassi Lomba de

Um estudo sobre adaptatividade dinâmica de dificuldade em jogos / Bruno Baère Pederassi Lomba de Araujo; orientador: Bruno Feijó. — Rio de Janeiro : PUC–Rio, Departamento de Informática, 2012.

v., 107 f: il. ; 30 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Informática – Tese. 2. Inteligência Artificial. 3. Modelagem de Jogador. 4. Jogos. 5. Adaptatividade Dinâmica de Dificuldade. 6. Ajuste Dinâmico de Dificuldade. I. Feijó, Bruno. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

Agradecimentos

À minha família e aos meus amigos pelo apoio, carinho e amor essenciais que não me deixaram fraquejar ante o desafio e por terem suportado minhas longas ausências.

Aos companheiros de pós-graduação do Departamento de Informática por terem sido verdadeiros irmãos-de-armas nessa batalha de 2 anos que todos enfrentamos.

Ao meu orientador Bruno Feijó pelo possível e impossível na orientação e por acreditar em mais esta aventura de um pesquisador que se recusa a abandonar a paixão pelos jogos.

À professora Simone Barbosa pelo inestimável apoio.

Aos membros da equipe V3O2 do TeCGraf pelo apoio e compreensão nas partes críticas deste trabalho.

À CAPES e à PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ter sido realizado.

Resumo

Araujo, Bruno Baère Pederassi Lomba de; Feijó, Bruno (Orientador). **Um estudo sobre adaptatividade dinâmica de dificuldade em jogos**. Rio de Janeiro, 2012. 107p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

As áreas de modelagem de jogador e sistemas adaptativos são relevantes para jogos ao permitir que o jogo se adapte ao nível de habilidade do jogador, procurando oferecer um desafio contínuo e de acordo com sua evolução de modo que o jogador se sinta motivado a continuar jogando pelo desafio proporcional ao seu desempenho. O presente trabalho realiza um estudo em modelagem de jogador e ajuste dinâmico de dificuldade, levantando dados sobre jogos que utilizam tais conceitos. Um sistema adaptativo baseado em modelagem de jogador é implementado e testado com jogadores para analisar a efetividade e relevância do uso de tais sistemas na manutenção do interesse do jogador, sob as óticas da teoria de fluxo (*flow theory*) e do modelo de elementos centrais da experiência de jogo (CEGE).

Palavras-chave

Inteligência Artificial ; Modelagem de Jogador ; Jogos ; Adaptatividade Dinâmica de Dificuldade ; Ajuste Dinâmico de Dificuldade.

Abstract

Araujo, Bruno Baère Pederassi Lomba de; Feijó, Bruno (Advisor).
A study on dynamic difficulty adaptivity in games. Rio de Janeiro, 2012. 107p. MsC Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The areas of player modeling and adaptive systems are relevant to games by allowing the game to adapt itself to the player's skill level, trying to offer a continuous challenge to the player, relative to its evolution in a way the player feels motivated to keep playing by the challenge proportional to his performance. The present work is a study on player modeling and dynamic difficulty adjustment, collecting data about games that use such concepts. An adaptive system based on player modeling is developed and tested with players to analyze the effectiveness and relevance of using such systems for maintaining the interest of the player, from the perspectives of the flow theory and the core elements of the gaming experience (CEGE) model.

Keywords

Artificial Intelligence; Player Modeling; Games; Dynamic Difficulty Adaptivity; Dynamic Difficulty Adjustment.

Sumário

1	Introdução	12
1.1	Motivação	12
1.2	Estado-da-arte	13
1.3	Nossa proposta	14
1.4	Estrutura da dissertação	15
2	Conceitos	17
2.1	Jogo	17
2.2	Jogador	24
2.3	Fluxo	28
3	Adaptatividade em jogos	32
3.1	Adaptatividade vs. Adaptabilidade	34
3.2	Exemplos comerciais de jogos adaptativos	35
3.3	Exemplos acadêmicos de jogos adaptativos	37
3.4	Jogos parcialmente adaptativos	39
3.5	<i>Framework</i> proposto por Charles e Black	42
3.6	Métodos adaptativos	43
3.7	Problemas da adaptatividade	44
3.8	Adaptatividade dinâmica ativa de dificuldade	46
4	Modelagem do jogador	48
4.1	Taxonomias de modelagem de jogador	49
4.2	Usos de modelagem de jogador	52
4.3	Técnicas	52
5	Metodologia	54
5.1	Tipo do jogo	54
5.2	Algoritmo para jogo adaptativo	54
5.3	Definição do jogo	58
5.4	Sistema adaptativo	61
5.5	Implementação da modelagem do jogador	64
5.6	Testes com jogadores	67
6	Análise dos resultados	77
6.1	Análise da população	77
6.2	Avaliação do sistema adaptativo	77
6.3	Resultados dos testes com jogadores	81
7	Conclusão e trabalhos futuros	85
7.1	Principais contribuições	86
7.2	Trabalhos futuros	86
	Referências Bibliográficas	88

A	Questionários	97
A.1	Pré-teste	97
A.2	Pós-experiência de jogo	98
A.3	Entrevista	100
B	Tabelas dos testes	101

Lista de figuras

1.1	<i>Switches</i> do Atari 2600	14
2.1	Classificação de jogos por gêneros na Playstation Store	22
2.2	Diagrama do canal de fluxo (<i>flow channel</i>)	29
2.3	As oito dimensões da experiência	30
3.1	Tela de ajuste de dificuldade de God of War	41
3.2	Tela de notificação de dificuldade do FIFA 12	41
3.3	<i>Framework</i> adaptativo de Charles e Black	42
3.4	Comparação entre DDA e Active DDA	46
3.5	Imagem do jogo fOw	47
4.1	Exemplo de modelo de jogador hierárquico	53
5.1	Exemplo de situação de jogo	55
5.2	Menu de seleção de dificuldade.	59
5.3	Tela de fim de jogo.	61
5.4	Tela de vitória.	62
5.5	Diagrama de classes UML do AIManager	63
5.6	Sobreposição de partes da implementação ao <i>framework</i> de Charles e Black	64
5.7	Elementos centrais da Experiência de Jogo	70
5.8	Relacionamento entre as variáveis do questionário CEGE	73
6.1	Participantes por sexo	78
6.2	Participantes por classe	78
6.3	Frequência de participantes por classe de idade	79
6.4	Frequência de gêneros conhecidos	79
B.1	População de testes	102
B.2	Resultado dos questionários de experiência de jogo para versão adaptativa	103
B.3	Resultado dos questionários de experiência de jogo para versão não-adaptativa	104
B.4	Recorte dos questionários para jogadores dedicados	105
B.5	Recorte dos questionários para jogadores casuais	106
B.6	Jogadores e desempenho nas versões	107

Lista de tabelas

2.1	Definições de jogo	21
2.2	Taxonomia dos jogos eletrônicos de Gularte	25
2.3	Oito elementos de Fluxo e os elementos de <i>Game-play</i> correspondentes	31
4.1	Taxonomia de modelagem de jogador segundo Machado et al.	49
4.2	Taxonomia inclusiva de Smith et al.	51
5.1	Modelos de jogador implementados	67
5.2	Relação das perguntas do questionário com os fatores da experiência de jogo	74
6.1	Resumo dos participantes - Sexo, Classe	77
6.2	Resumo dos participantes - Idade, Horas e Gêneros conhecidos	80
6.3	Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação ao total da população de testes	81
6.4	Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação aos jogadores dedicados	82
6.5	Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação aos jogadores casuais	83

... e todos sentiram que a aventura era muito mais perigosa do que haviam imaginado, que, o tempo todo, mesmo que conseguissem passar por todos os perigos da estrada, o dragão estaria esperando no fim.

J. R. R. Tolkien, *O Hobbit*.

1

Introdução

Recentes pesquisas nas áreas de *player modeling* e sistemas adaptativos (Charles04, Gilleade04, Charles05, Hunicke05) mostram que são campos de estudo relevantes e que ainda possuem desafios a serem resolvidos (Lopes11). Além de pesquisas acadêmicas, há casos conhecidos de adaptatividade na indústria, como o sistema de inteligência artificial *AI Director* de *Left 4 Dead* (Booth09), *Left 4 Dead 2* e *Max Payne* (Charles05).

Nesta dissertação, usamos os termos “sistemas adaptativos” e “adaptatividade” para representar sistemas e conceitos relativos a Ajuste Dinâmico de Dificuldade. Uma primeira razão para tal escolha de termos é porque eles revelam mais claramente a ideia de um jogo ir se adaptando ao jogador. A segunda razão é porque queremos explicitar a diferença entre “adaptatividade” e “adaptabilidade”, conforme apresentado na seção 3.1 desta dissertação.

1.1

Motivação

A pergunta que iniciou o interesse pela pesquisa em dificuldade adaptativa e modelagem de jogador foi “Por que o ser humano joga?”. Essa pergunta tem um forte cunho psicológico e social e é em Huizinga (Huizinga10) que encontramos a resposta: jogar é um fenômeno cultural que antecede a própria sociedade, sendo algo inerente ao ser humano.

“Encontramos o jogo na cultura, como um elemento dado existente antes da própria cultura, acompanhando-a e marcando-a desde as mais distantes origens até a fase de civilização em que agora nos encontramos.” (Huizinga10, p. 6)

Tornou-se necessário, portanto, repensar a pergunta, que passou a ser “Qual a motivação do ser humano em jogar determinado jogo?”. Para essa pergunta, que ainda possui uma conotação pertinente mais à Psicologia que às Ciências da Computação, Raph Koster (Koster04, pp. 38-42) mostra que jogos são como exercícios para o cérebro e a diversão surge do domínio de seus desafios. Dessa forma, enquanto o jogador for desafiado pelo jogo, haverá

interesse do jogador, mas deve haver um equilíbrio entre o desafio e a habilidade do jogador (Koster04, pp. 98 e 128). A teoria de fluxo de Csikszentmihalyi (Csikszentmihalyi90) auxilia essa abordagem da relação entre dificuldade e diversão em jogos, como mostramos na seção 2.3.1.

Dada essa relação entre dificuldade e diversão, a pergunta pode ser novamente reformulada, permitindo assim ser abordada pela área da Ciência da Computação: “Como podemos manter a motivação do jogador em jogar?”. Os trabalhos de (Charles04), (Charles05), (Hunicke05) e (Cowley08) mostram que a o ajuste dinâmico de dificuldade, ou adaptatividade dinâmica de dificuldade, aliado a uma modelagem de jogador, pode ser uma resposta para essa pergunta. Isso é corroborado pela afirmação de Koster (Koster04, p. 38), em que para tornar um jogo mais durável, em termos de rejogabilidade, é necessário integrar elementos de fora das mecânicas do jogo que tragam uma menor previsibilidade, como a psicologia humana, que pode ser considerada na modelagem de jogador.

O jogo Combat (Atari77), de 1977 para Atari 2600, possuía diversos modos de jogo que se diferenciavam, além do tipo (tanques e aviões), na dificuldade, oferecendo mais aviões e usando nuvens ou invisibilidade para esconder os jogadores ou aumentando o tamanho de um deles, por exemplo. Além disso, dois *switches* de dificuldade no corpo do console Atari 2600 (Figura 1.1) permitiam dar vantagens ou desvantagens aos jogadores para compensar diferenças na habilidade (Montfort09, pp. 31-32). Combat pode ser visto como o primeiro jogo, até onde os autores sabem, com mecanismo de ajuste de dificuldade, embora não dinâmico, sendo anterior à experiência de jogo e não durante esta, configurando como um jogo de dificuldade adaptável e não adaptativa, como definimos na seção 3.1.

1.2 Estado-da-arte

Como exemplo de jogos que implementam dificuldade adaptativa e são referência, temos Left 4 Dead e Left 4 Dead 2 (Valve08, Valve09, Booth09), que além de permitirem a escolha do nível de dificuldade que os jogadores desejam, realizam o ajuste do *spawn* de itens e inimigos baseado no desempenho dos jogadores, dessa forma realizando um ajuste fino da dificuldade ao oferecer itens que poderiam ajudar o grupo, como *medkits*, ou prejudicar através da inserção de hordas de zumbis e zumbis especiais para aumentar o desafio.

O jogo Max Payne (Gathering01) também é reconhecido pela sua implementação de um sistema de dificuldade adaptativa, ajustando o número de posicionamento dos inimigos na fase conforme o desempenho do jogador,



Figura 1.1: *Switches* do Atari 2600. Da esquerda para a direita: *power*, cor, dificuldade do jogador à esquerda, dificuldade do jogador à direita, seletor de jogo, *reset*. Fonte: Wikipedia, licenciada sob Creative Commons Attribution-Share Alike 3.0

embora critiquemos essa escolha em específico na seção 3.2.3 pelo impacto negativo que pode causar no reconhecimento de padrões do jogador.

1.3 Nossa proposta

Esta dissertação tem como proposta o estudo de técnicas de adaptatividade de jogos e técnicas de modelagem do jogador. O uso dessas técnicas visa manter o interesse do jogador ao proporcionar uma experiência de jogo que procure se adequar às características do jogador, em especial sua habilidade com aquele tipo de jogo.

Propomos também a implementação de um jogo utilizando as técnicas mencionadas acima, para poder analisar o impacto delas no desempenho e na satisfação do jogador.

Uma enumeração de nossos objetivos com este trabalho segue abaixo:

1. Realizar um estudo de técnicas de adaptatividade
2. Realizar um estudo de técnicas de modelagem de jogador
3. Implementar um jogo em duas versões com:
 - (a) dificuldade selecionável dividida em patamares (*Ex.: fácil, médio, difícil*)
 - (b) uma heurística de adaptatividade de dificuldade para inteligência artificial baseada em modelagem do jogador

4. Realizar testes de ambas as versões do jogo com jogadores para analisar a efetividade do uso de dificuldade adaptativa em comparação com dificuldade selecionável. A efetividade é avaliada em termos da satisfação do jogador em relação ao jogo através de questionários de avaliação de experiência de jogo.

É esperado que os jogadores prefiram o jogo com dificuldade adaptativa por ele poder se adequar às suas características, dentro de uma modelagem de jogador, proporcionando uma experiência melhor direcionada àquele jogador dentro das limitações do sistema.

A construção do sistema adaptativo do jogo está descrita no capítulo 5, seção 5.4. Os resultados encontram-se no capítulo 6.

1.4

Estrutura da dissertação

Esta dissertação foi organizada da seguinte forma:

Capítulo 1 – Introdução

Esta introdução. Detalha a motivação da dissertação, seus objetivos e sua organização.

Capítulo 2 – Conceitos

Neste capítulo abordamos três conceitos importantes para o entendimento deste trabalho: jogos, jogador e fluxo. Fazemos uma análise do conceito de jogo, enfatizando os jogos eletrônicos, a taxonomia proposta por Gularte (Gularte10) e uma análise do conceito de fluxo e sua aplicação em jogos.

Capítulo 3 – Adaptatividade em jogos

Neste capítulo, tratamos de um dos conceitos-chave deste trabalho: adaptatividade. Definimos a diferença entre adaptatividade e adaptabilidade. visitamos exemplos comerciais, acadêmicos e o *framework* proposto por Charles e Black. (Charles04).

Capítulo 4 – Modelagem de jogador

Outro conceito-chave deste trabalho é abordado neste capítulo: modelagem do jogador. Apresentamos a definição de modelagem do jogador de Cowley et al. (Cowley08), a taxonomia proposta por Machado et al. (Machado11a), usos e algumas técnicas.

Capítulo 5 – Metodologia

Este capítulo descreve a metodologia deste trabalho, da implementação

do jogo com sistema adaptativo ao método de avaliação. Conceitos específicos da metodologia são descritos neste capítulo.

Capítulo 6 – Análise dos resultados

Este capítulo apresenta os resultados obtidos com os testes realizados com os jogadores. Fazemos uma análise dos dados obtidos e apresentamos nossa interpretação dos resultados.

Capítulo 7 – Conclusões e trabalhos futuros

Neste capítulo, apresentamos as conclusões sobre a análise do trabalho realizado e seus objetivos e apresentamos problemas que pretendemos analisar em trabalhos futuros.

Apêndice A – Questionários

Este apêndice contém os questionários pré e pós-teste usados nos testes com jogadores.

Apêndice B – Tabelas por jogador

Este apêndice contém as tabelas com os dados coletados por jogador nos questionários pré e pós-teste e a evolução dos jogadores através das etapas dos testes.

2 Conceitos

Neste capítulo, apresentamos alguns conceitos necessários para delimitar o escopo deste trabalho e para sua melhor compreensão. Os conceitos de jogo e fluxo são limitados a jogos de *videogame*, que aqui entendemos não apenas como jogos exclusivos de consoles domésticos ou portáteis, mas também como jogos de computador e de *arcade*. A conceituação de jogador é importante para que possamos compreender seu papel no jogo e a importância de se considerar o jogador no *design* do jogo através da modelagem de suas características.

Apresentamos também uma classificação taxonômica para jogos que consideramos importante para ajudar a delimitar as questões discutidas neste trabalho a certas classes de jogos.

2.1 Jogo

2.1.1 Definição

Para Huizinga (Huizinga10), o jogo é uma função social que:

- Possui uma realidade autônoma (i.e., independe de outros conceitos para existir).
- É livre (uma atividade voluntária, que não pode ser imposta).
- É uma evasão da vida real, capaz de absorver inteiramente o jogador.
- É desinteressado.
- É limitado no espaço e tempo.
- Cria ordem e é ordem (i.e., possui regras).
- Exige um esforço dos jogadores para ser levado ao desenlace (esforço este causado pelo acaso ou incerteza).

“(...) se baseia na manipulação de certas imagens, numa certa ‘imaginação’ da realidade (ou seja, a transformação desta em imagens)(...)” (Huizinga10, p. 7)

“(...) se insinua como atividade temporária, que tem uma finalidade autônoma e se realiza tendo em vista uma satisfação que consiste nessa própria realização.” (Huizinga10, p. 12)

A definição de jogo também foi trabalhada posteriormente por outros autores, como Callois, Avedon & Sutton-Smith, Crawford e Salen & Zimmerman (Ranhel09, pp. 6-8). Essas novas definições foram compiladas em uma tabela (ver Tabela 2.1) por Jasper Juul (Juul03). A partir dos elementos comuns às definições, Juul (op. cit.) obteve a definição de jogo que adotamos para este trabalho:

“Um jogo é um sistema formal baseado em regras com um resultado variável e quantificável, onde a diferentes resultados são assinalados diferentes valores, o jogador exerce um esforço para influenciar o resultado, o jogador se sente emocionalmente ligado ao resultado e as consequências da atividade são opcionais e negociáveis.” (Juul03, tradução nossa).

A partir das seis características que definem o que é um jogo identificadas por Juul (op. cit.), é possível também, através da remoção de uma ou mais dessas características, classificar atividades em jogos, jogos fronteirços (ou quase-jogos) e não-jogos (Ranhel09, p. 12). Ranhel aponta um problema na definição de jogo de Juul (op. cit.) em que a partir do momento que se negocia as regras de um jogo, este sai do círculo do jogo, podendo tornar-se quase-jogo ou não-jogo (Ranhel09, p. 14). Porém, a definição de jogo de Juul (op. cit.) é adequada para jogos de *videogame* por se tratarem de sistemas cujas regras não são negociáveis (Ranhel09, p. 15), mesmo com uso de trapaças (*cheats*), pois seu uso já era previsto na implementação do jogo¹.

Koster (Koster04) contribui para a definição de jogo explicando o fator da diversão, apontado por Huizinga (Huizinga10, p. 5) como essencial ao jogo. Para Koster (op. cit., p. 40), a diversão nos jogos surge a partir da compreensão e do domínio dos desafios do jogo (vistos como padrões) e o tédio surge da necessidade de absorver novos padrões, quando o jogo não é mais capaz de fornecer variações nos padrões ou com dificuldade além ou aquém das capacidades do jogador (Koster04, p. 44).

Xavier identifica ainda duas outras características dos jogos, jogabilidade (Xavier10, p. 210) e anti-budismo (op. cit., p. 215), introduzido por Poole em

¹Aqui podemos abrir uma discussão sobre a negociação das regras ao realizar o *hacking* dos jogos para obter vantagens, como, por exemplo, através do uso de ferramentas como Game Genie e *patches* não oficiais. Entretanto, essa discussão não está no escopo deste trabalho.

Trigger Happy (Poole07, p. 103). Jogabilidade, também identificada por *playability* ou *game-play* em outros autores, como Salen e Zimmerman (Salen03), é definida por Xavier como:

“A jogabilidade, é portanto, uma decorrência natural do lidar com o jogo, como o termo usabilidade o é para com produtos e processos de manipulação. Jogabilidade como característica intrínseca ao conjunto de ações que são esperadas do jogador para com todos os seus componentes, sejam eles audiovisuais, sejam eles emocionais ou mesmo puramente cognitivos. O termo pode ser encarado ainda como a habilidade do jogador para com os mecanismos do jogo.” (Xavier10, p. 211)

E *game-play* é definido por Salen e Zimmerman como:

“...um processo reativo, delimitado por regras e emergente de sequências de ações pertencentes a um ou mais jogadores. Essencialmente, as regras governando a exploração do espaço de possibilidades do jogo por esse processo são as mecânicas de jogo: mecânicas influenciam pesadamente a experiência do jogador.” (Salen06, tradução nossa).

Em nossa análise, a jogabilidade é a interação do jogador com o jogo, mediada pelas regras do jogo implementadas na forma de mecânicas de jogo. Usamos, portanto, os termos jogabilidade e *game-play* como sinônimos neste trabalho, embora jogabilidade (*playability*) esteja mais relacionada a aspectos de usabilidade do jogo.

Para Xavier, o domínio do jogo e aprendizado através da repetição no jogo pode ser chamado de anti-budismo, que é definido (Xavier10, pp. 215-217) como o aprendizado através de sucessivas “reencarnações”, isto é, conforme o jogador erra (seu personagem morre ou precisa reiniciar o jogo), ocorre o aprendizado, sendo este aprendizado proporcional à quantidade de “reencarnações”. Esse aprendizado dá-se tanto na forma do domínio das mecânicas de jogo quanto na disposição dos seus elementos no tabuleiro (que pode ser real, como o tabuleiro do jogo de xadrez, ou virtual, como sua representação eletrônica ou uma fase do jogo Super Mario Bros.), correspondente à definição de jogo de Koster (op. cit.). O conceito foi originalmente introduzido por Poole (Poole07, p. 103) ao tratar o conceito de “vidas” em jogos como uma forma de budismo de ética reversa, onde ter mais vidas (reviver mais) é mais vantajoso.

“O jogador que aprende a perceber os padrões do inimigo pode inclusive, sacrificar uma de suas “vidas” para alcançar um conhecimento que residiria no futuro. Quando confrontado com o desafio novamente, sem sua segunda manifestação rediviva, saberá exatamente como proceder, numa espécie de gambito ético de proporções restritas. Morra e se lembre, diríamos: onde se errou? Como se movimentar com mais segurança na próxima vez? Como ampliar uma situação de contra-ataque na próxima vez? Onde encontrar elementos de aprimoramento de características normalmente básicas na próxima vez? O budismo acredita na transcendência da alma (controle-mater) e sua reencarnação constante em etapas que em algum momento, farão do indivíduo um ser iluminado cuja reencarnação seja desnecessária. A vida é uma etapa de sofrimento provador que em algum momento deixa de ser participada, pois a evolução foi alcançada. Quanto menor o número de reencarnações, melhor. No jogo, a idéia de morte colabora com um entendimento de aprendizado posterior que entre outros benefícios apresenta a possibilidade de se obter novas “vidas” para o estoque (mediante determinada pontuação). Vidas essas que serão ceifadas para que novas “vidas” e conhecimentos tácitos do jogo possam ser adquiridos. Portanto, quanto mais se “reencarna”, melhor!” (Xavier10, p. 216-217)

2.1.2

Taxonomias de jogos

Para Crawford (Crawford84, p.25), a classificação taxonômica dos jogos é importante para seu estudo por explicitar similaridades e diferenças entre famílias de jogos e entre membros de famílias, além de revelar princípios fundamentais de *game design*. Crawford estabelece um paralelo com o desenvolvimento da teoria da evolução de Darwin, surgida a partir de seus estudos sobre classificação dos seres vivos.

Crawford alerta que nenhuma classificação é totalmente correta, visto que diversas taxonomias podem ser definidas, considerando os diferentes aspectos dos jogos (e.g., mecânica e narrativa). Além disso, a evolução da área cria novos jogos que preenchem lacunas nas taxonomias anteriores, gerando assim a necessidade de novas taxonomias serem criadas para refletir tais mudanças (Crawford84, p.35).

Apperley aponta que as classificações usualmente aplicadas baseiam-se em aspectos estéticos dos jogos (Apperley06). Tal derivação decorre das

Tabela 2.1: Definições de jogo. Adaptada de (Ranhel09, p. 10)

Autores	Definição
Johan Huizinga, (Huizinga10, p. 16)	“...uma atividade livre, conscientemente tomada como ‘não séria’ e exterior à vida habitual, mas ao mesmo tempo capaz de absorver o jogador de maneira intensa e total. É uma atividade desligada de todo e qualquer interesse material, com a qual não se pode obter qualquer lucro, praticada dentro de limites espaciais e temporais próprios, segundo uma certa ordem e certas regras. Promove a formação de grupos sociais com tendência a rodearem-se de segredos e a sublinharem sua diferença em relação ao resto do mundo por meio de disfarces ou outros meios semelhantes.”
Roger Callois	“[o jogo] é uma atividade que é essencialmente: livre (voluntária), separada (no tempo e espaço), incerta, improdutiva, governada por regras, fictícia (faz-de-conta).”
Bernard Suits	“Jogar um jogo é se engajar em uma atividade dirigida para causar um estado específico de ocorrências, usando somente meios permitidos por regras, onde as regras proíbem meios mais eficientes, e onde tais regras são aceitas apenas porque elas tornam possível tal atividade.”
Avedon & Sutton-Smith	“No seu nível mais elementar, podemos definir jogo como um exercício de sistemas de controle voluntário, nos quais há uma oposição entre forças, confinado por um procedimento e regras, a fim de produzir um resultado não estável.”
Chris Crawford, (Crawford84, p. 7)	“Eu percebo quatro fatores comuns: representação [um sistema formal fechado, que subjetivamente representa um recorte da realidade], interação, conflito e segurança [o resultado do jogo é sempre menos severo do que as situações que o jogo modela].”
David Kelly	“Um jogo é uma forma de recreação constituída por um conjunto de regras que especificam um objeto (objetivo) a ser almejado e os meios permissíveis de consegui-lo.”
Salen & Zimmerman, (Salen03, p. 80)	“Um jogo é um sistema no qual jogadores engajam-se em um conflito artificial, definido por regras, que resultam em um resultado quantificável.”

primeiras classificações de jogos terem se baseado nas classificações de outras mídias (e.g., filmes), cuja principal diferença em relação aos jogos é o aspecto interativo.

O uso de taxonomias de jogos, além de importante para o aspecto de estudo do campo, tem uma importância para a venda de jogos. Ao separarmos e agruparmos jogos segundo determinadas classificações, permite-se que o consumidor e futuro jogador escolha o jogo baseado em comparações com jogos de uma mesma família e entre famílias diferentes. Um exemplo é a classificação por gêneros da Playstation Store da Sony, Figura 2.1, que divide os jogos em: ação, aventura, jogos de cartas, PS Eye², luta, *game show*³, Playstation Move⁴, música e ritmo, jogos online, plataforma, *puzzle*, corrida, *retro*⁵, RPG, *shooters*, *shooting*⁶, simulação, esporte, estratégia, únicos⁷ e jogos 3D (não exibido na Figura 2.1).



Figura 2.1: Classificação de jogos por gêneros na Playstation Store. Fonte: própria.

Taxonomia de Crawford

A taxonomia de Crawford divide os jogos em dois principais grupos: jogos de ação e habilidade e jogos de estratégia (Crawford84, pp. 25-35). Crawford

²Jogos que usam a câmera PS Eye da Sony.

³Jogos de perguntas e resposta, estilo *quiz*.

⁴Jogos que fazem uso do periférico de captura de movimento da Sony.

⁵Jogos que fazem referência a jogos antigos, como os de *arcade*, em especial da década de 80 e início de 90.

⁶Jogos de tiro em primeira ou terceira pessoa.

⁷Jogos com mecânicas que se destacam pela diferença de outros gêneros mais consagrados, como *flOw* (thatgamecompany06).

admite que sua classificação toma como base aspectos históricos da aparição e reconhecimento dos gêneros de jogos, não se baseando em outros fatores.

Jogos de ação e habilidade exigem como principais habilidades do jogador a coordenação visual-manual e tempo de reação rápida. Subdividem-se em: combate, labirinto, esportes, raquete, corrida e diversos. Por razões históricas, jogos de raquete se referem a jogos derivados de Pong (Atari72) que usam a mecânica de uma raquete rebatendo, enquanto jogos diversos se referem a novas variações em jogos de *arcade* e consoles que teriam seu gênero estabelecido posteriormente, como Donkey Kong (Miyamoto81), que poderia ser classificado como plataforma em uma classificação contemporânea.

Jogos de estratégia se diferenciam dos jogos de ação e habilidade por não terem a mesma exigência de habilidade e coordenação do jogador. Subdividem-se em: aventura, D&D, jogos de guerra, jogos de azar, educacionais e interpessoais. Por jogos em estilo D&D, Crawford se refere a jogos que possuem inspiração em mecânicas de RPGs como Dungeons & Dragons (Gygax74), substituindo a figura do mestre de jogo pelo computador. Por jogos interpessoais, Crawford se refere a jogos que explorariam relacionamentos entre indivíduos ou grupos, inexistentes em computadores ou consoles na época da formulação de sua taxonomia. Hoje, podemos associar a essa categoria os MMORPGs, que inserem em suas mecânicas a criação de grupos e guildas.

Taxonomia de Gularte

Gularte (Gularte10) buscou construir uma classificação taxonômica para os jogos de videogame a partir dos trabalhos de Crawford (Crawford84, p. 25-35) e Rollings & Adams (Rollings03), baseada na jogabilidade, no gênero da história do jogo e na quantidade de jogadores, que apresentamos na Tabela 2.2. Embora não concordemos totalmente com a classificação proposta⁸ e os exemplos⁹, essa classificação é útil para analisar e comparar jogos eletrônicos. A taxonomia proposta por Gularte admite que um jogo possa ter múltiplas combinações taxonômicas para melhor identificar o público-alvo, as

⁸A classificação de jogos de tiro envolve jogos com características muito distintas que mereceriam subcategorias, enquanto jogos de estratégia possuem subcategorias que poderiam ser categorias à parte, como por exemplo RPG. Além disso, não há divisão entre jogos 2D e 3D, quanto à câmera (terceira pessoa, primeira pessoa, visão *top-down*, isométrica, etc.) e jogos de mundo aberto. Tais classificações ajudariam a delimitar melhor as especificidades de determinados jogos.

⁹Final Fantasy VII como um jogo de estratégia por turnos, quando o sistema utilizado é o *active battle system* em que não há espera do término do turno do oponente para que o NPC ou personagem do jogador atue, sendo o tempo entre cada ação determinado por uma barra de tempo, consistindo em um meio-termo entre tempo real e por turno. Second Life como jogo, quando está mais para uma ferramenta de comunicação com uso de avatares 3D.

principais características do jogo e seus elementos visuais, auditivos e sensoriais (Gularte10, p. 165).

2.1.3

Dificuldade

Para os jogadores habituais, pode ser difícil tentar explicar um conceito tão intrínseco aos jogos e dissociá-lo da separação em níveis comumente aceita (e.g.: *easy*, *normal* e *hard*). O conceito de dificuldade é derivado da relação entre habilidade e desafio: quanto maior a habilidade necessária para resolver o desafio, maior a dificuldade. Esta relação é extraída do conceito de fluxo, apresentado na seção 2.3.

Como para cada ser humano o conceito de dificuldade é relativo às suas habilidades e ao desafio projetado, a dificuldade é subjetiva. Torna-se, portanto, difícil comparar a dificuldade de um jogo, pois a pergunta passa a ser “Difícil em relação a quem ou a quê?”. A comum divisão de dificuldade de um jogo em pataranes ou níveis representa a visão do *designer* do que é desafiador para os tipos de jogadores com habilidades para os quais os desafios foram projetados.

Na seção 2.3, analisamos como a dificuldade a par das habilidades é desejada por determinados jogadores através do entendimento do conceito de fluxo de Csikszentmihalyi (Csikszentmihalyi90) e no capítulo 3 como podemos proporcionar o canal de fluxo com o ajuste dinâmico de dificuldade.

2.2

Jogador

Uma vez definido jogo, cabe definir o jogador. A conceituação do jogador é importante para entendermos seu relacionamento com o jogo, ou seja, o que busca ao jogar o jogo, o que o motiva a jogar e quais são os fatores do jogo que amplificam a experiência com dado jogador. Enquanto no capítulo 4 estamos vendo o jogador sob o ponto de vista da correlação de suas características com as mecânicas de jogo, nesta seção vemos o jogador de forma mais abstrata.

A identificação do jogo como elemento cultural feita por Huizinga (Huizinga10) considerada na seção 2.1 coloca o jogador como um agente que busca a diversão no jogo. Embora jogos possam ser usados como forma de treinamento (Schell11, p. 117) por sua capacidade de abstrair a realidade em modelos e padrões mais facilmente absorvidos (e.g., o xadrez como abstração da movimentação de exércitos em guerra), a diversão é essencial ao jogo e quando não se faz presente, torna-se desinteressante para o jogador. Essa busca pela

Tabela 2.2: Taxonomia dos jogos eletrônicos de Gularte. Adaptada de (Gularte10)

Classificação	Exemplos
Quanto ao estilo	
Jogos de ação e habilidade	
Luta	Street Fighter II, Mortal Kombat
Tiro	Space Invaders, Quake 3
Plataforma	Super Mario Bros., Castlevania
Jogos de estratégia	
Turnos	Civilization
RPG	Diablo, Final Fantasy
Explore, expand, exploit, exterminate	Starcraft, Civilization
Simuladores	
Transportes	Flight Simulator
Militares	America's Army, Ghost Recon
Construção	Sim City, Zoo Tycoon
Sociabilização	The Sims, Second Life
Esportes	
Corrida	Pole Position, F1 2010
Coletivos	FIFA Soccer, NBA Jam
Individuais	Virtua Tennis, Chessmaster
Educativos	
Matérias escolares	Coelhinho Sabido
Conhecimento	Carmen Sandiego, Show do Milhão
Treinamento (<i>Serious games</i>)	
Jogos específicos	
Adulto	Leisure Suit Larry
Arcade	Teenage Mutant Ninja Turtles 2
Advergame	
Artilharia	Virtua Cop
Música	Guitar Hero, DDR
Quebra-cabeça	Lemmings, Pipe Mania
Pinball	Pinball Dreams
Filmes interativos	Dragon's Lair
Labirinto	Pac-Man, Berzerk
Quanto ao gênero	
Aventura	Super Mario Bros., God of War
Guerra	Call of Duty, Medal of Honor
Adulto	
Ação	Street Fighter II, Need for Speed
Terror	Silent Hill, Resident Evil
Policial	Grand Theft Auto, Swat 3
Fantasia	Final Fantasy, Spyro
Infantil	Turma da Mônica, Pac-Man
Quantidade de jogadores	
<i>Single player</i>	Uncharted, Sonic
<i>Multi player</i>	Contra, World of Warcraft

diversão no reconhecimento e domínio de padrões em um jogo é identificado por Koster (Koster04, p. 40,44) como o que motiva o jogador a jogar.

As diferentes motivações que levam os jogadores a preferirem determinados jogos ou a exibirem determinados comportamentos dentro de jogos permitem-nos classificá-los quanto a essas motivações, como vemos na seção 2.2.1.

2.2.1

Classificação de jogadores

Casuais e dedicados

Uma recente forma de se classificar os jogadores adotada pela comunidade de desenvolvedores¹⁰ é dividi-los entre casuais e dedicados¹¹ (Novak11, p. 54).

Jogadores casuais são considerados aqueles que jogam ocasionalmente, por vezes não se considerando jogadores, optando por jogos rápidos que não exijam dedicação e que proporcionem uma distração rápida, optando tipicamente por jogos de celular ou via *web*. Jogadores dedicados são considerados aqueles que dedicam mais que apenas algumas horas por semana aos jogos, optando tipicamente por jogos mais complexos, que possuem narrativas e mecânicas mais elaboradas ou demoradas. Para Fortugno:

“Jogadores casuais não abordam jogos com o mesmo conjunto de habilidades que jogadores dedicados e portanto possuem níveis diferentes de exploração automotivada e paciência com falhas.”

(Fortugno08, p. 144, tradução nossa)

Os jogadores dedicados estão mais acostumados com a exposição a determinadas convenções em jogos, como formas de controle do avatar e da câmera, além de mecânicas de jogos comuns. Por estarem mais acostumados a explorar e experimentar em jogos, são mais tolerantes a frustrações com falhas e com dificuldades de assimilar conhecimento (Fortugno08, p. 146).

Entretanto, os jogadores casuais não estão acostumados às mesmas convenções que os jogadores dedicados nem estão condicionados a tolerar falhas. A noção dessa diferenciação é importante para o projeto de interfaces,

¹⁰Embora Novak considere que casuais e dedicados sejam extremos que definem uma frequência de jogar, a mídia especializada e a comunidade de jogadores adotou essa classificação como forma de se separar os jogadores “tradicionais” dos “novos jogadores”, que antes não jogavam jogos eletrônicos tradicionais em computadores ou consoles de *videogame* e foram inseridos neste contexto pela popularização de jogos em celulares e sistemas que contemplaram este nicho de não-jogadores, como o Wii da Nintendo (Fortugno08, p. 143–144).

¹¹*Hardcore*.

desafios e mecanismos de *feedback* para jogos que contemplem jogadores casuais (Fortugno08, p. 147–148).

Classificação demográfica

Novak propõe que os jogadores sejam observados sob a ótica da demografia (Novak11, pp. 56–70), tanto pelas diferenças quanto ao sexo, quanto pelas diferenças culturais da geração dos jogadores, sendo divididos entre: geração silenciosa (de 1924 a 1943), *baby boomers* (de 1943 a 1961), geração X (de 1962 a 1981) e geração do milênio (de 1982 a 2002). As observações de Koster quanto à idade e ao interesse por jogos (Koster04, pp. 4–10, 48–50) também indicam que a faixa etária é um fator a ser considerado na classificação do jogador. Schell identifica o grau de interesse por jogos por faixa etária (Schell11, pp. 100–102) e elenca diferenças na preferência de temas em jogos por homens e mulheres (Schell11, pp. 102–108).

Psicotipos

Tradicionalmente, tipologias de jogadores como a proposta de Bartle (Bartle96), (Fullerton08, p. 51), (Cowley08), (Novak11, p. 40), (Schell11, pp. 110–112), (Stewart11), (Mallory12) e como as tipologias psicológicas de Myers-Briggs e psicotipos VALS (Novak11, pp. 54–55) são usadas para classificar jogadores quanto aos aspectos motivacionais que impactam na experiência de jogo. Jogos são projetados pelos *designers* com determinados tipos de experiências como foco e tipologias como as mencionadas, além de *frameworks* de *design* que visam a experiência como o MDA (Hunicke04a) e o EGE (Kultima10) servem como ferramentas para auxílio ao *design* do jogo considerando a experiência do jogador. Como disseram Koster, “Está claro que jogadores tendem a preferir certos tipos de jogos que parecem corresponder às suas personalidades.” (Koster04, p. 104, tradução nossa) e Schell, “É útil examinar esses diferentes prazeres porque diferentes pessoas atribuem diferentes valores a cada um.” (Schell11, p. 110).

Na seção 2.3, a relação entre motivação e experiência de jogo é analisada sob a ótica da teoria de fluxo, sobre a otimalidade da experiência. Nos capítulos 3 e 4 analisamos o uso de técnicas e ferramentas para compreender o jogador e adaptar a experiência de jogo àquele em específico.

2.3

Fluxo

Faz-se necessário definir fluxo¹² para que possamos correlacionar a dificuldade de um jogo com a diversão proporcionada ao jogador. O trabalho de Cowley et al. (Cowley08) faz um extenso estudo de fluxo e sua relação com jogos, usando conceitos como modelagem de jogador para explicitar a importância dessa relação.

“Fluxo é um sentido que os humanos desenvolveram para reconhecer padrões de ação que valem ser preservados e transmitidos pelo tempo. Este foi o maior *insight* de Huizinga: que as instituições ‘sérias’ que constituem a sociedade – ciência, a lei, as artes, religião e até mesmo as forças armadas – todas começaram como jogos, em um contexto em que as pessoas podiam jogar e experimentar a diversão de uma ação direcionada a um objetivo.” (Csikszentmihalyi90, tradução nossa).

Para Schell (Schell11, p. 118), fluxo é “uma sensação de foco completo e energizado em uma atividade, com alto nível de prazer e satisfação”. Segundo Cowley et al. (Cowley08), a teoria de fluxo diz que é possível para qualquer pessoa experimentar fluxo em qualquer atividade, contanto que os seguintes pré-requisitos sejam atendidos: as habilidades da pessoa sejam compatíveis com os desafios da atividade na proporção correta e que a pessoa tenha uma personalidade autotélica (i.e., seja capaz de reconhecer e perseguir oportunidades de fluxo). Além disso, atingir o estado de fluxo requer um considerável investimento de tempo e esforço em uma atividade. A Figura 2.2 mostra como desafio e habilidade são correlacionados com fluxo. A Figura 2.3 mostra outras experiências que o participante da atividade pode ter dada a variação de habilidades e desafios.

O estado de fluxo é caracterizado pelos seguintes fenômenos: foco extremo na tarefa; sensação de controle ativo¹³; misto de ação e percepção; perda da auto-percepção (noção de identidade); distorção da experiência de tempo; a experiência da tarefa ser a única justificativa necessária para continuá-la executando;

Schell (Schell11, p. 119) coloca como alguns dos principais componentes para se ter uma atividade que coloque o participante em estado de fluxo:

“

¹²Optamos por utilizar a forma traduzida, fluxo, ao invés do original *flow* por seu uso anterior em (Schell11).

¹³Alguns autores se referem a esse fenômeno como agência.

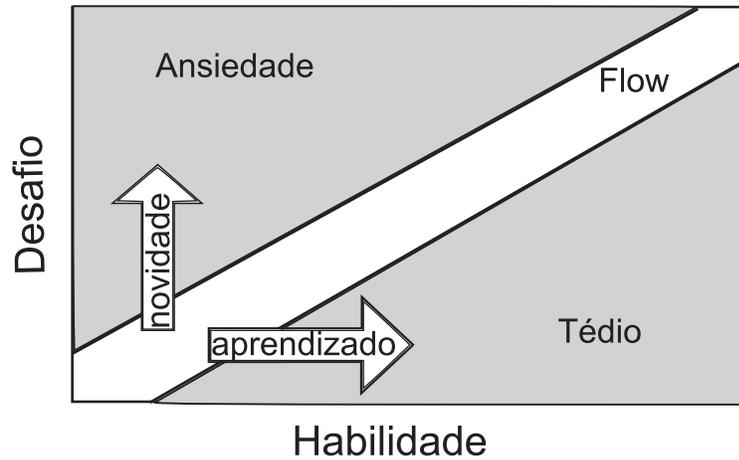


Figura 2.2: Diagrama do canal de fluxo (*flow channel*). Adaptada de (Cowley08)

Objetivos claros: Quando nossos objetivos são claros, somos capazes de nos manter mais facilmente focados em nossa tarefa. Quando os objetivos não são claros, não focalizamos nossa tarefa, uma vez que não estamos certos se nossas ações atuais são úteis.

Sem distrações Distrações roubam o foco de nossa tarefa. Sem foco, sem fluxo.

Feedback direto: Sem cada vez que tomarmos uma ação, tivermos de esperar antes de conhecermos o efeito que a ação provocou, ficaremos rapidamente distraídos e perderemos o foco na nossa tarefa. Quando o *feedback* é imediato, podemos nos manter facilmente focados.

Desafio contínuo: Seres humanos adoram um desafio. Mas ele deve ser um desafio que achamos que podemos alcançar. Se começarmos a pensar que não conseguiremos alcançá-lo, nós nos sentiremos frustrados e nossa mente começa a procurar uma atividade que possa ser recompensadora. Por outro lado, se o desafio for muito fácil, nós nos sentiremos entediados e, novamente, nossa mente começa a procurar atividades mais recompensadoras.

” (Schell11, p. 119).

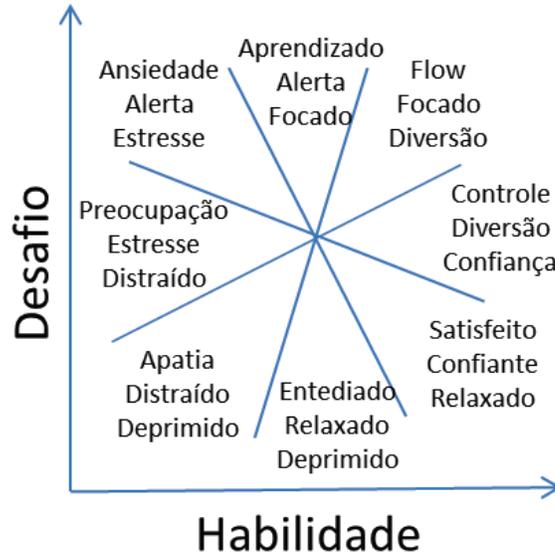


Figura 2.3: As oito dimensões da experiência. Adaptada de (Cowley08)

2.3.1

Conceito de fluxo em jogos

Cowley et al. (Cowley08) entendem fluxo, no contexto de jogos, como um estado de operação ótimo em um sistema de informações, do ponto de vista de que jogos são atividades limitadas por regras que o jogador aceita ao escolher fazer parte da atividade. Definem ainda para este propósito sistemas de informações como qualquer sistema complexo que toma uma entrada, processa e gera uma saída, realizando uma analogia com o ato de jogar, em que jogos recebem uma entrada do jogador e retornam o resultado do processamento dessa entrada. Como um exemplo simples, podemos tomar o pressionar de um botão pelo jogador em um controle e a consequência desta ação, como o avatar¹⁴ do jogador pular. Cowley et al. citam ainda Csikszentmihalyi (Csikszentmihalyi90), relacionando fluxo e reconhecimento de padrões, que Koster (Koster04, pp. 38-42) definiu como parte importante para a sensação de diversão em um jogo. Fluxo recebe uma lente¹⁵ própria na obra de Schell (Schell11), enfatizando a importância de sua consideração no processo de *game design*.

¹⁴Avatar é uma palavra do sânscrito que significa encarnação e desde Ultima IV de Richard Garriot é usada como sinônimo da representação do jogador no universo do jogo, não estando limitado a jogos de videogame (Montfort09, p. 51). Exemplos de avatares vão desde peões em jogos de tabuleiros a modelos 3D animados em jogos de videogame modernos.

¹⁵Em “A Arte de Game Design” (Schell11), Schell caracteriza sua experiência em *game design* sob a metáfora de “lentes” que auxiliam o designer a observar o jogo sob diferentes aspectos.

Tabela 2.3: Oito elementos de Fluxo e os elementos de *Game-play* correspondentes. Adaptada de (Cowley08)

Elementos de Fluxo	Elementos de <i>Game-play</i>
Uma tarefa desafiadora mas possível de ser completada	A experiência de jogo completa (incluindo interação social durante o jogo)
Imersão completa na tarefa, sem intromissão de outras preocupações	Alta motivação para jogar, sem obrigação do contrário, empático ao conteúdo
Sentimento de controle total	Familiaridade/habilidade com o controle, convenções do gênero, mecânicas de jogo
Liberdade total para se concentrar na tarefa	Telepresença e um ambiente dedicado ao jogo
A tarefa possui objetivos claros e sem ambiguidade	Missões, linha do enredo, níveis: qualquer resultado explícito de uma sessão de jogo
Resposta imediata nas ações	Penalidades de contingência e recompensas adequadas e no momento certo
Ter menos noção da passagem do tempo	Foco em outro ambiente independente de tempo
Sensação de identidade diminui, mas é logo reforçada	Incorporação do avatar, sensação de conquista após o jogo – e.g. “Hi-score”

Na pesquisa original de Csikszentmihalyi sobre fluxo, como lembram Cowley et al. (Cowley08), os participantes experimentaram fluxo quando havia um equilíbrio entre desafios e habilidade. Como Koster (Koster04, p. 38-42) notou, diversão em jogos depende desta sensação de desafio e o reconhecimento de padrões, os jogadores se divertem mais jogando quando atingem o estado de fluxo. Cowley et al. compilaram a relação entre os oito elementos de fluxo e elementos de *game-play* na Tabela 2.3.

Cabe ainda considerar as conclusões do trabalho de Cowley et al. que fornecem o embasamento para o uso de adaptatividade em jogos: tendo considerado o jogo como um sistema de informações, o desafio é proporcional à taxa de recepção de informações e a habilidade é proporcional ao desempenho frente o desafio e o potencial ótimo de desempenho, os autores propõem que um sistema construído para dinamicamente ajustar as entidades de jogo que fornecem informações, i. e. desafio, de modo que o *game-play* seja adaptado para a ótima experiência do jogador, a fim de que o jogo seja reativo ao jogador através de aspectos tangíveis de sua experiência, mediados por uma heurística derivada do construto de fluxo.

3 Adaptatividade em jogos

Adaptatividade em jogos não é um tema recente: Missura e Gärtner citam a utilização de adaptatividade em jogos de tabuleiro como as vantagens oferecidas ao outro jogador em Go e no esporte golf (Missura11); No campo dos jogos eletrônicos, o Departamento de Registro de Patentes dos Estados Unidos possui o registro de uma patente (Morrison81) para um jogo adaptativo controlado por um microcontrolador, de modo que o jogo consiga detectar o nível de habilidade dos participantes para ajustar a dificuldade de acordo com o maior nível detectado. O jogo Zanic (Compile86a) para NES é reconhecido (Thomas07) como um dos primeiros jogos de *videogame* a ter um sistema de dificuldade adaptativa implementado, o “Automatic Level of Difficulty Control” (Compile86b, p. 4), que modificava a dificuldade do jogo de acordo com o desempenho do jogador.

Sistemas adaptativos têm sido usados para dar ao jogador uma experiência de jogo personalizada, ajustando o jogo direcionado por objetivos que podem ser identificados, mensurados e influenciados (Lopes11), geralmente definidos por um fator dinâmico, como por exemplo a perícia do jogador, que evolui conforme ele joga ou o número de jogadores em uma equipe. O sistema adaptativo realiza o ajuste baseado nestes objetivos, seja adaptando a dificuldade ou outros fatores como NPCs¹, disposição de itens para se adequarem aos recursos do jogador (Booth09, Hunicke05), etc. Com isso, os sistemas adaptativos procuram alcançar o equilíbrio do jogo para cada jogador. Por essa razão, alguns autores usam o termo balanceamento ou equilíbrio de jogo, como por exemplo Andrade et al. (Andrade06). Schell identifica doze diferentes tipos de equilíbrios em jogos (Schell11, pp. 171–205).

Equilíbrio em um jogo é definido por Novak (Novak11, p. 202) como um estado em que os jogadores percebem que o jogo é consistente, justo e divertido. É dividido em equilíbrio estático, relativo às regras do jogo (mecânicas e suas dinâmicas) e equilíbrio dinâmico, relativo à interação do jogador com o jogo. Para um jogo estar equilibrado, Novak determina que a jogabilidade deve

¹*Non player characters*: Personagens não jogadores, elementos do jogo, geralmente adversários ou personagens de apoio controlados por inteligência artificial.

fornecer:

1. Desafios compatíveis com a habilidade do jogador.
2. Uma experiência de jogo justa, não sendo o jogador condenado desde o princípio por causa de seus “erros”.
3. Ausência de estagnação, o jogador não pode ficar sem rumo.
4. Ausência de decisões banais, de forma que o jogador deva ser responsável apenas por decisões importantes.
5. Níveis de dificuldade, podendo o jogador escolher ou a dificuldade se ajustar à habilidade do jogador durante o jogo.

Cabe notar que os itens 1, 3 e 5 encontram-se de acordo com o que se espera de uma experiência que possa proporcionar fluxo ao jogador, conforme observamos na seção 2.3.

Para Andrade et al. (Andrade06), a adaptatividade dinâmica em jogos deve satisfazer três requisitos: identificar e se adaptar ao nível do jogador o mais rápido possível; perceber e registrar as mudanças de desempenho do jogador; na adaptação, o comportamento do jogo deve se manter verossímil e discreto, de forma que o jogador não perceba o sistema.

Segundo Lopes e Bidarra (Lopes11), a adaptatividade pode ser realizada de duas formas: *offline*, que é o ajuste usando dados do jogador realizado antes da jogabilidade ativa, geralmente durante a inicialização ou durante o carregamento; *online*, realizada no jogo em tempo real, conforme o jogo ocorre.

Porém, há ainda alguns desafios a serem resolvidos, como apontado por Lopes e Bidarra (Lopes11). Por exemplo, para adaptar o jogo conforme a motivação do jogador em jogar, é necessário: determinar a expectativa do jogador; quantificá-la de forma que possa ser medida, comparada e adaptada; e realizar o processamento dos sinais para que os ajustes corretos sejam realizados no jogo. Os dois primeiros itens são problemas de modelagem de jogador, que abordamos no capítulo 4, e o terceiro item é um problema de adaptação. Outro desafio levantado por Lopes e Bidarra é em dar suporte aos mecanismos de adaptatividade de modo que sejam reutilizáveis independentemente do domínio² do jogo.

Neste trabalho, as seguintes considerações são feitas em relação à adaptatividade em jogos. Consideramos como jogos:

²Por domínio do jogo, entendemos que Lopes e Bidarra (Lopes11) se referiam aos diferentes estilos e gêneros de jogos, como indicado na Tabela 2.2 e as especificidades relativas a cada um.

Adaptativos: aqueles que oferecem um mecanismo de ajuste do jogo ao jogador aumentando e diminuindo a dificuldade automaticamente de forma a facilitar ou dificultar o progresso do jogador conforme seu desempenho.

Parcialmente adaptativos: aqueles que oferecem: ou uma forma automática de detecção de que o jogador está com habilidade acima da dificuldade atual, mas não realiza a mudança automaticamente; ou um escalamento da dificuldade em apenas uma direção, seja apenas aumentando ou apenas diminuindo a dificuldade até atingir um patamar compatível com o jogador.

Não-adaptativos: aqueles que não oferecem um mecanismo adaptativo de dificuldade.

Cabe ainda explicitar que a variação de dificuldade proporcionada pelos jogos adaptativos e parcialmente adaptativos está relacionada com a evolução do jogador (em termos de habilidades e conhecimento para vencer os desafios propostos pelo jogo) e/ou de seu avatar (relacionado ao nível de poder do avatar, em termos de mecânicas de jogo, de vencer os desafios). É esperado que os jogos forneçam naturalmente por *design* um aumento de dificuldade gradual com a evolução do jogo, não configurando esta como uma característica adaptativa. Por exemplo, é esperado que os níveis mais avançados de Super Mario Bros. sejam mais difíceis que os níveis iniciais, refletindo o aprendizado ao longo do jogo projetado pelo designer, mas não é esperado que os desafios se ajustem ao estilo de jogo do jogador e ao seu aprendizado específico. Essas diferenças são melhor explicitadas na seção 3.1.

3.1

Adaptatividade vs. Adaptabilidade

Por adaptatividade, queremos dizer a capacidade do sistema se adaptar ao usuário, ao contrário de adaptabilidade, que se refere à capacidade do sistema ser modificado por um usuário ao seu desejo (Gilleade04). Jogos que permitem a seleção de parâmetros (e.g. quantidade de vidas inicial, dificuldade, fogo-amigo habilitado) são considerados jogos adaptáveis³ e geralmente apresentam uma evolução fixa e progressiva no desenrolar do jogo em relação aos desafios enfrentados pelo jogador, visto como um usuário ideal, com aprendizado linearmente crescente em relação ao jogo. Jogos adaptativos são aqueles

³Houlette classifica tais jogos como manualmente adaptativos (Houlette04, p. 557). Não usamos essa nomenclatura pelas razões apresentadas anteriormente nesta mesma seção.

que apresentem uma mudança dinâmica em sua jogabilidade, seja esta mudança aplicada na inteligência dos NPCs ou nos desafios gerados, refletindo a interação do jogador com o jogo e oferecendo uma experiência única a cada jogador.

Gilleade et al. (Gilleade04) criticam essa pressuposição de uma evolução fixa de dificuldade dentro dos parâmetros estabelecidos no início dos jogo adaptáveis, atentando ao fato de que os jogadores podem não se enquadrar no ideal projetado pelo game designer para aquela experiência. Como apontado por Koster (Koster04, p. 38-42), o jogador precisa encontrar desafio e reconhecer e dominar padrões no jogo para se divertir, e um jogo que não acompanhe o aprendizado do jogador pode se tornar frustrante.

3.2

Exemplos comerciais de jogos adaptativos

3.2.1

Left 4 Dead

O exemplo mais recente de jogo adaptativo é a série Left 4 Dead da Valve (Valve08, Valve09), com seu sistema de inteligência artificial AIDirector (Booth09). Segundo a taxonomia de Gularte (Tabela 2.2), Left 4 Dead é classificado como um jogo de tiro de ação *multi player* (se o número de quatro jogadores não for satisfeito, os restantes são controlados por uma inteligência artificial). Os jogadores selecionam a dificuldade da fase antes de começar a jogar e o sistema AIDirector realiza a distribuição de inimigos (e.g. hordas de zumbis, zumbis especiais como Tank, Smoker, etc.) e de itens, como escopetas e *medkits*, de acordo com o estado dos personagens em cada zona da fase, usando a dificuldade selecionada previamente como direcionador. Booth (Booth09) esclarece que o jogo é visto como um desafio de habilidade, ao contrário de um exercício de memorização, usando a geração procedural parcialmente aleatória para prover uma imprevisibilidade estruturada que promove a rejogabilidade e a experiência projetada pelo designer.

3.2.2

Mario Kart 64

Outro exemplo de adaptatividade é o jogo Mario Kart 64 da Nintendo (Nintendo96), classificado segundo a taxonomia de Gularte como um jogo de corrida de ação e *single* ou *multi player*. Para permitir que jogadores de maior e menor habilidade pudessem encontrar desafios adequados, a inteligência

artificial do jogo utiliza uma técnica que é conhecida como “rubber banding”⁴ (Charles04, Charles05, Hunicke05), que ajusta a velocidade dos NPCs e probabilidade dos itens para ajudar ou atrapalhar o jogador de acordo com sua posição na corrida. Entretanto, essa técnica pode ser abusada pelo jogador, mantendo-se em uma posição menor ou jogando deliberadamente de forma inferior à sua capacidade para se aproveitar do ajuste. Esse é um dos problemas da adaptatividade dinâmica que tratamos na seção 3.7.

3.2.3

Max Payne

Em Max Payne (Gathering01), classificado como um jogo de ação de tiro *single player* na taxonomia de Gularte, Lopes e Bidarra (Lopes11) apontam um mecanismo adaptativo que é usado para manipular a dificuldade alterando o auxílio à mira do jogador e mesmo a quantidade e disposição dos NPCs. Entretanto, segundo a definição de Koster para a diversão em jogo (Koster04, p. 40) que enunciamos no capítulo 2, seção 2.1, a diversão com o jogo pode ser prejudicada ao alterar a quantidade e disposição dos elementos que exigem domínio e compreensão ao se ajustar dinamicamente a dificuldade. Por exemplo, morrendo⁵ diversas vezes ao tentar invadir uma sala com muitos inimigos em determinada posição, o jogador se sentirá enganado ao perceber o inimigo que se escondia atrás da porta não está nem mesmo em outro lugar da sala quando recarregar o *savepoint*⁵, pois seu conhecimento espacial adquirido com a experiência enfrentará uma situação que não era esperada, contrariando a característica do anti-budismo dos jogos.

3.2.4

GundeadliGne

GundeadliGne (RockinAndroid10) é um jogo de tiro de aventura *multi player* pela taxonomia de Gularte, pertencente ao subgênero *bullet hell*, onde há diversos padrões de tiros que preenchem a tela e inimigos que o jogador deve evitar que atinjam um ponto pequeno centrado em seu avatar, da largura de alguns pixels.

⁴Elástico, em uma tradução livre. A analogia deve-se ao fato do ajuste ser feito para equilibrar jogadores que se afastem muito, tanto para as primeiras quanto últimas posições, tentando arrastá-los para o “meio”.

⁵*Savepoint* é um ponto do jogo onde o progresso do jogador é salvo para que possa retornar caso seu personagem morra ou precise abandonar o jogo. Não são todos os jogos que oferecem essa possibilidade, sendo pertencente geralmente aos jogos de ação e habilidade (luta, tiro, plataforma) e em alguns jogos específicos (artilharia, filmes interativos) – segundo a classificação de Gularte apresentada na Tabela 2.2 – embora possa-se perceber que há uma tendência nos atuais jogos de incluí-la.

Além de poder escolher a dificuldade inicial, o jogo apresenta um sistema adaptativo que controla a densidade de tiros na tela e sua velocidade chamado de *phase*. A variação do *phase* vai de 1 a 21 e é limitada pelo nível de dificuldade escolhido, aumentando conforme o jogador elimina inimigos e mantém o botão de tiro rápido pressionado e diminuindo conforme o jogador perde vidas, usa bombas ou captura determinados itens.

Jogos do estilo *bullet hell* são reconhecidos como muito difíceis, exigindo dedicação para serem dominados. O uso de um sistema de dificuldade adaptativa permite que jogadores de menor habilidade ou que não estejam acostumados com o estilo possam jogá-lo sem entretanto enfrentar uma dificuldade proibitiva, além de permitir um aumento no desafio para jogadores acostumados com o estilo.

3.2.5 Retro/Grade

Retro/Grade é um jogo da 24 Caret Games a ser lançado na PSN⁶ na data da escrita deste trabalho. Seus autores incluíram um sistema de adaptatividade dinâmica de dificuldade através de uma seleção de quais *powerups* serão oferecidos ao jogador dependendo de seu desempenho na fase⁷.

3.3 Exemplos acadêmicos de jogos adaptativos

3.3.1 Infinite Adaptive Mario

O jogo Infinite Adaptive Mario foi criado por Weber (Weber10a) a partir de uma modificação do Infinite Mario Bros de Persson (Persson08)⁸, utiliza o criador de mundos procedimental para gerar os desafios (i.e., posicionamento de inimigos e geografia da fase) baseado no nível de dificuldade atual do jogador, em uma escala de 1 a 100. O ajuste é realizado baseado em quão bem o jogador conseguiu avançar naquele mundo, indo gradativamente para dificuldades maiores e reduzindo a dificuldade consideravelmente quando há

⁶<http://blog.us.playstation.com/2012/07/27/retrograde-blasts-off-on-psn-august-21st>

⁷<http://www.indiepubgames.com/news/designing-difficulty-in-retro-grade-indie-game>

⁸Infinite Mario Bros foi criado por Markus Persson para uma competição de jogos em Java baseados em Super Mario Bros, de Shigeru Miyamoto (Nintendo85), sendo classificado segundo a taxonomia de Gularte como um jogo de aventura de plataforma *single player*. Por gerar a fase de maneira procedimental, o jogo é utilizado em competições de inteligência artificial, onde os participantes devem criar um *bot* que consiga navegar pela fase gerada por mais tempo antes de morrer.

mortes seguidas. Uma versão anterior do jogo recebeu o primeiro lugar da competição de jogos da IEEE CIG 2010 na trilha de geração de fases (Weber10b, Weber10c).

Embora o jogo consiga realizar a adaptatividade da dificuldade, oferecendo ao jogador níveis mais difíceis ou mais fáceis dependendo de seu desempenho, basear a dificuldade do mundo na disposição dos elementos de *level design* prejudica a imersão do jogador por se tratar de mudanças mais facilmente perceptíveis, isto é, o jogador tem a noção de que o jogo o está ajudando quando tem um mau desempenho modificando a fase, enquanto o bom desempenho (sucesso em chegar ao objetivo) o recompensa com uma nova fase com um nível de dificuldade maior proporcional ao tempo que levou para alcançar o final da fase.

3.3.2

Hamlet

Hamlet (Hunicke04b, Hunicke05) é uma modificação do jogo Half-Life da Valve (Valve98), um jogo de tiro de ação *single player*. A adaptatividade envolve verificar a condição do mundo de jogo (i.e., a disposição de elementos como itens e inimigos) em relação ao desempenho do jogador, ajustando o mundo de jogo sob uma política com base econômica (i.e, baseada no consumo e demanda de recursos) norteada pelo *framework* de *game design* MDA (Hunicke04a), que correlaciona as mecânicas de jogo, seu comportamento emergente na forma de dinâmicas e a experiência estética projetada.

Conforme o jogador avança no jogo vencendo os desafios, as recompensas tornam-se mais variadas, escarças e valiosas, fazendo com que o jogador tenha que se empenhar mais em obtê-las com a variação da dificuldade. O paralelo com o *framework* MDA se dá no suporte à estética dos jogos de tiro em primeira pessoa (geralmente classificados como jogos de tiro de ação na taxonomia de Gularte), através das dinâmicas da variação das recompensas, como munição e itens de recuperação de saúde do avatar, e da variação dos desafios enfrentados, que fazem parte das mecânicas do jogo.

Uma implementação similar pode ser conferida em Half-Life 2 (Valve04), onde os itens de recuperação de saúde encontrados em caixas após um segmento de conflito durante o jogo variam de acordo com a saúde do avatar no momento.

3.3.3

Adaptive Pong

Ibáñez e Delgado-Mata (Ibáñez11) desenvolveram um jogo similar a Pong (Atari72) usando ajuste dinâmico de dificuldade para manter o interesse de

ambos os jogadores, facilitando o jogo para o jogador com baixo desempenho, para quem o jogo está difícil, e dificultando para o de melhor desempenho, para quem o jogo está fácil, variando parâmetros como comprimento e velocidade da raquete de cada jogador e velocidade da bola dependendo do jogador que a acerta. O objetivo dessa alteração nas mecânicas de jogo é promover a interação social entre os jogadores apesar da diferença de nível de habilidade entre eles.

Em seu trabalho, Ibáñez e Delgado-Mata identificaram que o mecanismo de adaptatividade dinâmica favoreceu o interesse no jogo ao evitar o tédio causado em um jogador de maior perícia e ao evitar a frustração em um jogador com menor perícia quando estes se confrontam, visto que para um o desafio é menor que sua habilidade e para o outro o desafio está além. Esses resultados estão de acordo com a teoria de fluxo, mostrando que é possível usar dificuldade dinâmica adaptativa para manter o interesse de jogadores em jogos *multi player*.

3.4

Jogos parcialmente adaptativos

Nesta seção discutimos brevemente duas séries de jogos que implementam parcialmente um mecanismo adaptativo. Por jogos parcialmente adaptativos, queremos dizer aqueles em que ou a dificuldade não é ajustada dinamicamente, apenas verificando que o jogador poderia estar em outro nível de dificuldade, ou a dificuldade é ajustada em apenas uma direção, geralmente aumentando, mas não diminuindo a partir do ponto de referência (dificuldade inicial), chamado por Novak e Dunningway de escalamento ou rampa de dificuldade (Dunningway08, p. 223-228). Consideramos tais implementações como parcialmente adaptativas por não oferecerem ambas as possibilidades de aumento e/ou redução de dificuldade dinamicamente sem interferência do jogador.

3.4.1

Série Mega Man Zero

A série Mega Man Zero (IntiCreates02, IntiCreates03, IntiCreates04, IntiCreates05) pode ser classificada como jogos de aventura de plataforma *single player* na taxonomia de Gularte. Na série, dependendo da forma como as fases são completadas pelo jogador, uma pontuação é atribuída, classificando em um *ranking* de F (pior desempenho) a S (melhor desempenho). Essa pontuação influencia na forma como o chefe de cada fase lutará, podendo usar técnicas especiais que podem ser obtidas pelo jogador, embora não haja uma redução na dificuldade caso o *rank* do jogador na fase seja muito baixo.

3.4.2

The Elder Scrolls IV: Oblivion

O jogo Oblivion (Bethesda06), classificado como um jogo de RPG de aventura *single player*, apresenta um mecanismo parcialmente adaptativo ao escalar a dificuldade do jogo de acordo com o nível⁹ do personagem do jogador. Novak e Dunniway (Dunniway08, p. 226) acreditam que a utilização dessa dificuldade escalada com o nível do personagem se deva à dificuldade e aos custos de criação de conteúdo para jogos, valendo-se desse mecanismo para criar variedade nos desafios.

Porém, desse modo, um jogador que desvie da missão principal logo em seu início para evoluir seu personagem realizando missões alternativas¹⁰, encontrará desafios que não condizem com o momento do desenvolvimento da história do jogo. Como o sistema é baseado apenas no nível do personagem, cuja evolução em níveis depende do uso de habilidades, sem considerar o equipamento do mesmo, os desafios gerados podem se tornar impossíveis de serem vencidos sem que o jogador interrompa a missão para procurar um equipamento adequado.

Outro problema derivado deste uso de adaptatividade é a modificação da fauna do jogo para se adequar ao personagem, podendo tornar raros encontros com NPCs de nível menor que são necessários em determinadas missões, além da quebra de verossimilhança do mundo de jogo quando qualquer bandido de estrada encontrado possui equipamento similar ao do jogador.

3.4.3

God of War e FIFA 12

Embora não apresente diretamente um sistema de dificuldade adaptativa, a série God of War (Sony05) verifica a frequência de mortes do jogador e oferece para diminuir a dificuldade do jogo (Figura 3.1), que pode ser reajustada a qualquer momento. O jogo FIFA 12 (EAMobile11) oferece ao jogador a possibilidade de modificar o nível de dificuldade ao analisar a quantidade de gols marcados ou sofridos (Figura 3.2).

Enquanto a possibilidade de reajustar a dificuldade pode ser desejável para alguns jogadores, para outros a mensagem para diminuir a dificuldade, embora com a intenção de ajudar, pode ser interpretada como ofensiva, como se o jogo duvidasse da capacidade do jogador de superar aquele desafio. Um

⁹Em jogos de RPG, uma possível forma de medir o poder do personagem é através do nível. O conceito é amplamente utilizado em outros jogos que absorveram características dos jogos de RPG e deriva dos jogos de RPG de mesa, como Dungeons & Dragons (Gygax74).

¹⁰Optamos por “missões alternativas” como tradução para *side quests*, embora “missões paralelas” também possa ser utilizado.

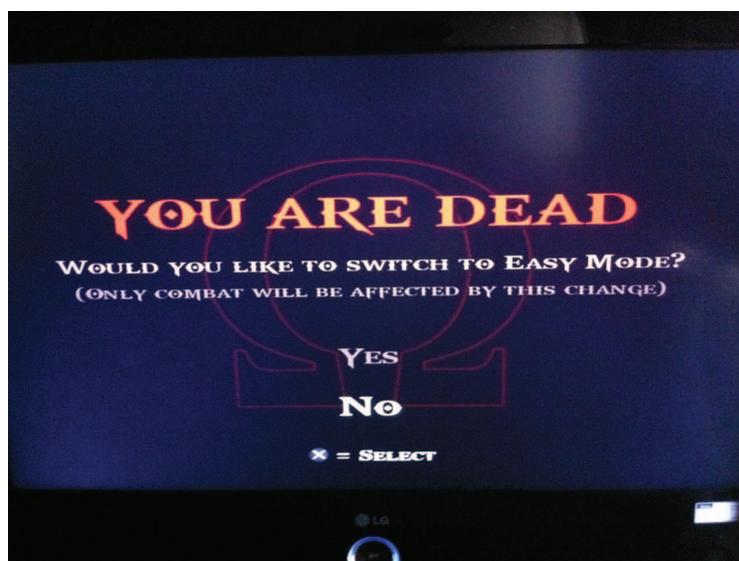


Figura 3.1: Tela de ajuste de dificuldade de God of War. Fonte: própria.

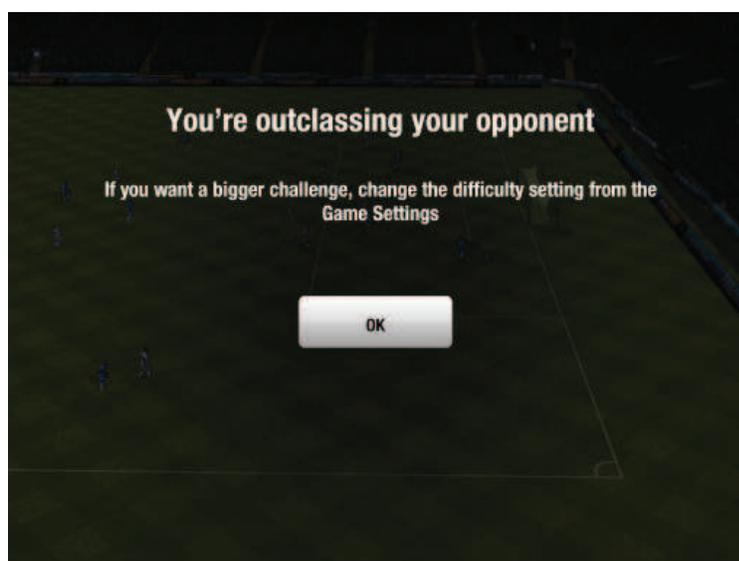


Figura 3.2: Tela de notificação de dificuldade do FIFA 12. Fonte: própria.

ajuste pequeno e imperceptível na dificuldade sem a necessidade de aumentar a quebra de imersão causada pela mensagem na tela (afinal, ao oferecer para diminuir a dificuldade, estamos lembrando ao jogador de que aquilo é “apenas um jogo”) poderia trazer um resultado melhor, com uma possibilidade maior de que o jogador acreditasse que foi somente por seu mérito e dedicação que o desafio foi vencido.

3.5 Framework proposto por Charles e Black

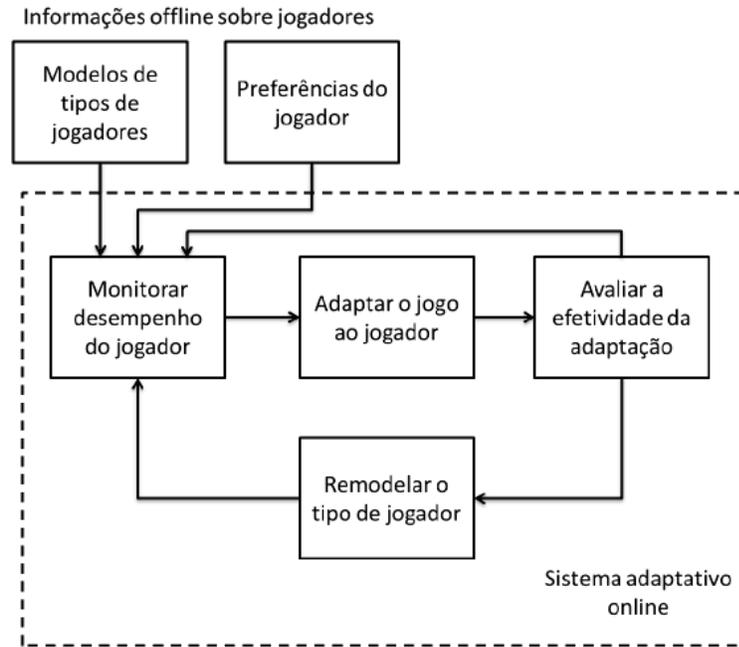


Figura 3.3: *Framework* adaptativo de Charles e Black (Charles04).

A Figura 3.3 mostra a proposta de *framework* de Charles e Black (Charles04) e Charles et al. (Charles05) para um sistema de dificuldade adaptativa. Sua principal contribuição foi a formalização da necessidade de uma modelagem de jogador acoplada ao desempenho adaptativo do jogo, dessa forma permitindo que a efetividade da adaptação dos agentes inteligentes seja medida através da evolução do modelo do jogador, correlacionando com o nível de frustração do mesmo. Se não há detecção de avanço do desempenho do jogador ou diminuição de sua frustração, o jogador pode ter sido incorretamente classificado ou a modelagem que correspondia inicialmente ao jogador já não é mais válida. Modelagem de jogador, suas classificações e métodos são melhor analisados no capítulo 4.

O *framework* de Charles e Black é composto de quatro aspectos principais: modelagem de jogador, ambiente de jogo adaptativo em resposta às necessidades do jogador, monitoramento da efetividade ou compatibilidade de qualquer adaptação e remodelagem ou classificação dinâmica de jogador. O *framework* recebe como entrada inicial uma modelagem de jogador que pode ter sido feita através de processos computacionais, como o uso de redes neurais, ou através de definições de *design* do jogo e um conjunto de preferências

do jogador, como por exemplo o tipo de jogo preferido, dificuldade inicial e demais parâmetros que possam classificá-lo dentro de uma modelagem.

3.6

Métodos adaptativos

Nesta seção, apresentamos uma pesquisa de métodos e modelos de adaptatividade, divididos conforme a técnica principal do método: ajuste de parâmetros, métodos comportamentais (e.g. scripting dinâmico) e técnicas de aprendizado de máquina (e.g. redes neurais e algoritmos genéticos). Cabe ressaltar que a maioria dos métodos adaptativos depende de alguma forma de modelagem do jogador para direcionar a adaptatividade.

3.6.1

Ajuste de parâmetros

Ibáñez e Delgado-Mata (Ibáñez11) realizam uma avaliação do desempenho do jogador através de uma função de avaliação que considera a pontuação de um jogador sobre o outro e o tempo entre cada ponto e realizam alterações necessárias nas características da raquete de cada jogador, como tamanho, deslocamento e velocidade de rebote. Essa função é chamada de função heurística por alguns autores (Andrade06) por dependerem de heurísticas específicas de cada jogo.

Na modificação de Half-Life, Hamlet (Hunicke04b, Hunicke05), Hunicke e Chapman avaliam o desempenho do jogador, sua probabilidade de morrer e o consumo de itens de seu inventário para aplicar políticas de ajuste

3.6.2

Aprendizado de máquina

Algoritmos evolutivos podem ser usados para adaptar as táticas dos NPCs ao jogador, como nos trabalhos de Demasi e Cruz, Johnson e Noon (Demasi03a, Demasi03b, Johnson04, Noon09). A seleção do *pool* genético para a próxima geração é feita selecionando os candidatos que melhor se aproximaram do nível do jogador. Noon usou controladores neuroevolutivos para gerar oponentes baseados em uma modelagem de jogador.

Andrade et al. (Andrade06) usam uma técnica de aprendizado reforçado. Os agentes inteligentes são condicionados a aprenderem o máximo possível mas são limitados a realizarem ações dentro de um conjunto equiparável mas não muito superior ao do jogador. Essa abordagem é similar ao método de Spronck et al. (Spronck06).

Moffett (Moffett10) aplicou uma rede neural para adaptar o jogo segundo seu modelo causal de diversão que é baseado em trabalhos como o MDA de Hunicke (Hunicke04a). O modelo apresenta três camadas, uma genérica, uma voltada a determinados gêneros de jogos e outra específica do jogo onde é utilizado, para poder avaliar a diversão do jogador e realizar os ajustes.

3.6.3

Métodos comportamentais

Spronck et al. usaram aprendizado de máquina através da técnica de *scripting* dinâmico (Spronck06). Essa técnica de gera *scripts* (conjuntos de regras comportamentais) dinamicamente para novos oponentes, usando uma base de regras geradas manualmente com um fator de probabilidade de serem inseridas no script modificado por um peso que é ajustado de acordo com o a taxa sucesso de seu uso, procurando manter um jogo desafiador. Machado et al. (Machado11b) apresentam um exemplo de uma arquitetura de aprendizado usando um classificador M5P. Ambos os trabalhos usam *top culling*, onde regras que tenham um peso maior que um limite máximo ajustado dinamicamente (i.e., que tiveram um desempenho muito acima do jogador) são descartadas para evitar a criação de oponentes com dificuldade muito acima do nível do jogador percebido pelo algoritmo, buscando assim usar regras que se aproximem do nível do jogador.

Outros trabalhos em adaptatividade e ajuste dinâmico incluem: Yannakakis e Hallam (Yannakakis08a) que usam redes neurais para modelar as preferências e comportamentos do jogador e modelos de entretenimento baseados nos fatores de Malone para *gameplay* (Malone84), utilizando os dados de entrada da interface física para realizar o ajuste dos parâmetros do jogo; Missura e Gärtner (Missura11) formalizam o ajuste dinâmico como um problema de predição em conjuntos parcialmente ordenados.

3.7

Problemas da adaptatividade

Uma das dificuldades em se usar adaptatividade em jogos, citada por Gilleade et al. (Gilleade04) é a necessidade de se considerar a motivação dos jogadores (por quê eles querem jogar), sua a experiência e as habilidades (quão bem conseguem jogar) e quando há a necessidade de mudança no *gameplay* através da adaptatividade (e.g., um jogador que deseja desafio pode se sentir frustrado ao ter a dificuldade reduzida). Esses questionamentos também são identificados em (Cowley08), embora direcionados à compreensão da experiência de jogo. Para Gilleade et al., uma possível solução seria detectar

a frustração dos jogadores, medida através do progresso no jogo e de eventos de entrada de dados e biométricos, como a pressão ao apertar botões no controle, a entrada de sequência de teclas que não fazem sentido, a força com que segura o controle e como o manipula.

Outro problema conhecido da adaptatividade dinâmica é a possibilidade de abuso do jogador do sistema em seu favor (Schell11, p. 205). Como citamos anteriormente na seção 3.2.2, a técnica de “rubber banding” implementada em Mario Kart 64 pode ser abusada pelo jogador para conseguir uma vantagem competitiva ao reduzir a velocidade dos competidores controlados pela inteligência artificial. Esse tipo de abuso pode ocorrer quando o jogador percebe o uso do sistema adaptativo e prejudica a imersão no jogo ao ter a noção de que os adversários estão dando uma vantagem. Esse problema também é apontado por Schwab (Schwab04, p. 153) no jogo Zanac para NES de 1986, de forma que é necessário filtrar comportamentos maliciosos ou estranhos do jogador para que sistema adaptativo não seja sabotado.

Embora não possa necessariamente ser considerado um abuso do mecanismo adaptativo, Charles e Black (Charles04) apontam o problema causado pela troca de jogadores durante o jogo, quando um passa a vez ao outro sem trocar o perfil de jogador ativo (caso algum tipo de perfil de jogador seja usado). Dessa forma, não há como obter um modelo de jogador e uma adaptação corretos.

Como citamos anteriormente na seção 3.4, o jogo Oblivion apresenta um mecanismo de dificuldade adaptativa que acaba gerando desafios que escalam baseados apenas no nível do personagem, causando a percepção de que o mundo evoluiu junto. Além dos citados anteriormente, alguns dos problemas decorrentes disso são: a quebra da verossimilhança (e.g., o sumiço de parte da fauna, trocados por animais e monstros de nível mais alto); a falta das sensações de poder e domínio sobre o desafio, necessárias para que a experiência de fluxo seja alcançada como visto na seção 2.3.1. Schell (Schell11, p. 205) também aponta este problema no tocante à quebra da imersão, quando o jogador percebe que as habilidades de seus inimigos não são absolutas, mas relativas ao seu desempenho, quebrando a ilusão do desafio fixo.

Por fim, um outro problema no uso de dificuldade dinâmica é que alguns jogadores podem se sentir insultados ou decepcionados quando percebem que o sistema está procurando facilitar seu avanço no jogo (Schell11, p. 205).

3.8

Adaptatividade dinâmica ativa de dificuldade

Para Chen (Chen06), apenas a automatização do ajuste dinâmico de dificuldade não é capaz de levar os jogadores a experimentarem fluxo, podendo ser inclusive prejudicial ao desenvolvimento da experiência por sua não-trivialidade de implementação e por tomar o controle de decisões que caberiam ao *game designer*. Chen sugere que os sistemas adaptativos normalmente implementados são passivos, reagem ao desempenho do jogador e não consideram sua particularidade de canal de fluxo. Um dos pontos citados por Chen é que um jogador pode experimentar fluxo ao levar seu avatar a atitudes suicidas, enquanto para o sistema de dificuldade adaptativa a contagem de mortes do avatar seria um indicador de baixo desempenho.

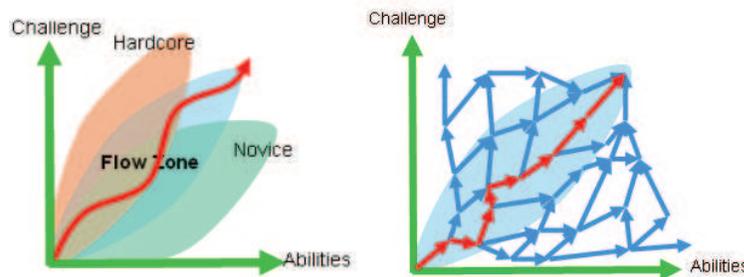


Figura 3.4: Comparação entre DDA e Active DDA, adaptado de (Chen06).

Chen propõe um sistema ativo de dificuldade dinâmica adaptativa, visando o fluxo, onde as decisões de adaptatividade são implementadas pelo *game designer* como escolhas de *gameplay*, delegando ao jogador a decisão de quando e como manipular a dificuldade. A implementação de referência desse sistema é demonstrada no jogo *fOw*¹¹ (thatgamecompany06) (Figura 3.5), onde o jogador toma a decisão de quando avançar ou retroceder na dificuldade, indo para o próximo nível ou retrocedendo um nível. A Figura 3.4 compara o diagrama de fluxo usual com o diagrama de fluxo da adaptatividade dinâmica ativa de dificuldade de Chen, onde o jogador influi na sua própria navegação pelo canal de fluxo através de suas escolhas e a seu ritmo.

¹¹<http://www.jenovachen.com/flowingames/flowing.htm>

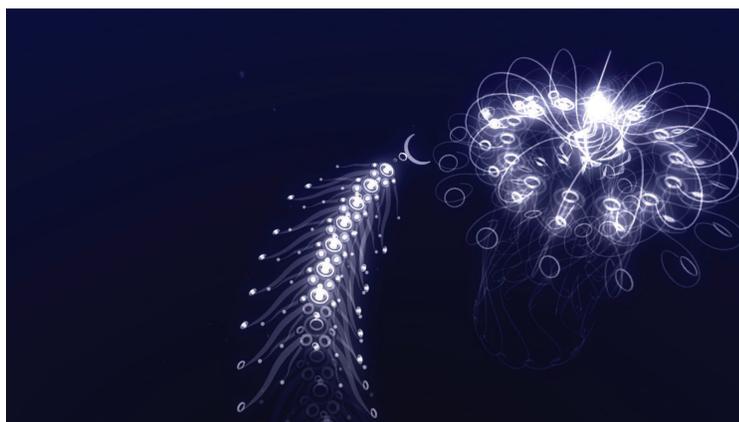


Figura 3.5: Imagem do jogo flOw. Fonte: thatgamecompany

4 Modelagem do jogador

Essa técnica também é conhecida como modelagem de adversários ou oponentes, como usada por alguns autores (Herik05, Bakkes09). Baseada no conceito de modelagem de estudantes usado em sistemas adaptativos educativos (Houlette04, p. 557), onde o programa de ensino detecta características do estudante para melhor direcionar seus estudos, a definição de Cowley et al. para *player modeling* é:

“[...] uma técnica para inferir atributos de alta-ordem do jogador usando uma larga quantidade de informações de dados de tipos primitivos do jogador e descrevê-lo a outros processos usando algum algoritmo de classificação”. (Cowley08, tradução nossa).

Cowley et al. (Cowley08) identificam que, durante uma sessão de jogo, o jogador é definido por informações pessoais e/ou dados do *log* de jogo, podendo ser modelado com construtos tipológicos (em relação às informações pessoais obtidas pelo jogo) e/ou construtos de modelagem (em relação aos dados da sessão de jogo). Informações com relevância semântica para a modelagem de jogador incluiriam dados de entrada e saída do jogo, como velocidade de reflexo do jogador, taxa de acertos, posicionamento em relação a oponentes e a média de repetições em uma fase, ou também dados externos, como classificação tipológica psicológica segundo algum modelo como o de Bartle (Bartle96) ou o modelo estendido de Stewart (Stewart11).

Tais informações coletadas do jogador podem ser usadas para adaptar a inteligência artificial do jogo ao jogador (Houlette04, p. 557), convertidas em características pertencentes a uma modelagem. Dependendo da perspectiva utilizada para verificar e classificar essas informações, elas podem servir como uma medida do conhecimento e habilidade do jogador ou como um indicador de seus hábitos e preferências, o que modifica a construção da modelagem, embora as diferenças entre uma e outra possam ser sutis. A complexidade do modelo em ser projetado, analisado e atualizado está diretamente relacionada à quantidade de detalhes do modelo, embora uma modelagem mais detalhada possa fornecer informações mais relevantes sobre o comportamento do jogador.

Tabela 4.1: Taxonomia de modelagem de jogador segundo Machado et al. (Machado11a)

Descrição	Categorias	Objetivos
Conhecimento	Rastreamento <i>online</i>	Colaboração
Posição	Reconhecimento <i>online</i> de estratégia	Adversarial
Estratégia	Revisão <i>offline</i>	Narrativa
Satisfação		Substituição
Aplicações	Métodos	Implementação
Especação em busca	Modelagem de ação	Explícita
Tutorial	Modelagem de pre-ferências	Implícita
Treinamento	Modelagem de posição Modelagem de conhecimento	

Pesquisas em modelagem de jogador incluem os trabalhos de Charles et al. em um *framework* de modelagem de jogador (Charles04, Charles05), o uso de aprendizado supervisionado para modelagem e ajuste dinâmico de dificuldade de Missura e Gärtner (Missura09) e o uso de redes bayesianas para geração de modelos e a avaliação do interesse dos jogadores por Yannakakis e Maragoudakis (Yannakakis05).

4.1

Taxonomias de modelagem de jogador

Nesta seção, apresentamos duas taxonomias propostas para modelagem de jogador: a proposta de Machado et al. (Machado11a) e a proposta de Smith et al. (Smith11).

4.1.1

Taxonomia de Machado et al.

Na taxonomia de Machado et al. (Machado11a), a modelagem de jogador pode ser classificada segundo seis fatores, resumidos na Tabela 4.1: tipo de descrição do jogador, categoria de uso da modelagem, objetivo da modelagem, aplicação da modelagem, método de modelagem e implementação da modelagem.

Quanto à descrição do jogador, a modelagem pode ser classificada: quanto ao conhecimento do jogador dentro do jogo (e.g. o quanto do mapa de jogo foi explorado); quanto ao posicionamento do jogador no mundo de jogo; quanto à satisfação do jogador em jogar (i. e. suas preferências de jogo); e quanto às estratégias que usa.

Quanto às categorias de uso, pode ser classificada: como rastreamento *online*, quando a preocupação é com as ações futuras imediatas do jogador; como reconhecimento *online* de estratégia quando envolve identificar um conjunto de ações como estratégias; e como revisão *offline* quando usa dados de fora da experiência de jogo, como *logs* de partidas anteriores.

Quanto aos objetivos, pode ser classificada como: colaborativa, quando o objetivo é gerar agentes que auxiliem o jogador; adversarial, quando o objetivo é gerar oponentes adaptados ao jogador; narrativa, quando o objetivo é ajudar na criação, manutenção ou avanço de uma linha narrativa (e.g. reações e emoções de NPCs).

Quanto à aplicação, pode ser classificada como: especulação em busca, quando a modelagem pode ser usada para auxiliar heurísticas de busca (e.g. poda α - β); tutorial, quando o uso da modelagem é voltado ao ensino do jogador por um agente inteligente; treinamento, quando o agente inteligente usa a modelagem para selecionar desafios voltados às características do jogador; substituição, quando o agente inteligente é usado para substituir um jogador humano (e.g. em *Left 4 Dead* (Valve08), quando não há jogadores suficientes para completar a equipe ou quando um dos jogadores perde a conexão).

Quanto ao método de modelagem, pode ser classificada como: modelagem de ação, quando a modelagem é feita sobre as ações do jogador, em uma tentativa de prever as ações seguintes; modelagem de preferência, quando a modelagem se refere aos objetivos do jogador com o jogo (o que ele pretende durante aquela experiência de jogo); modelagem de posicionamento, quando a modelagem usa informações de posições previamente conhecidas do jogador para articular uma estratégia (e.g. em jogos de estratégia para simular a ‘neblina de guerra’); modelagem de conhecimento, quando a modelagem é usada para inferir o conhecimento do jogador.

Quanto à implementação da modelagem, ela pode ser classificada como: explícita, quando o modelo de jogador existe em separado de sua representação em código dentro do jogo (e.g. através de *scripts* externos); implícita, quando as informações do modelo do jogador estão diluídas pela representação em código, sem se encontrar separadas e já computadas.

4.1.2

Taxonomia inclusiva de Smith et al.

A modelagem de jogador nesta taxonomia é dividida em quatro facetas: o escopo de aplicação; o propósito do uso; o domínio das características modeladas; e a fonte da derivação do modelo ou da motivação. A Tabela 4.2 sintetiza as classificações possíveis por faceta.

Tabela 4.2: Taxonomia inclusiva de Smith et al. (Smith11)

Escopo	Propósito	Domínio	Fonte
Individual	Geradora	Ações de jogo	Induzida
De classe	Descritiva	Reações humanas	Interpretada
Universal			Analítica
Hipotética			Sintética

Quanto à faceta do escopo, a modelagem pode ser: individual, quando os modelos se referem a jogadores específicos, permitindo comparações entre jogadores (não se restringindo apenas a jogadores humanos); de classe, quando se referem a categorias de jogadores, particionando o espaço de jogadores em estereótipos; universal, quando o modelo se aplica a todos os jogadores (um exemplo citado pelos autores é o de uma função de mensuração de diversão aplicada aos jogadores sem distinções específicas); hipotética, quando o modelo é projetado baseado em um tipo de jogador em específico através de suposições sobre seu comportamento.

Quanto à faceta do propósito, a modelagem pode ser: geradora, quando o modelo gera dados que poderiam ser adquiridos a partir de um jogador (por exemplo, na criação ou reprodução de determinados comportamentos para simular jogadores); descritiva, quando seu intento é fornecer informações sobre o jogador para diversos usos (por exemplo, um mapa de locais frequentados no jogo).

Quanto à faceta do domínio, a modelagem pode ser em relação a: ações de jogo, quando se refere a ações do jogador dentro do jogo (e.g., personalização do avatar, escolhas de movimentação, *upgrades*, etc.); reações humanas, quando se refere a propriedades da experiência de jogo externas à realidade simulada do jogo (e.g., dados de movimentação ocular do jogador, nível de diversão ou frustração do jogador).

Quanto à faceta da fonte da derivação, a modelagem pode ser: induzida, quando o modelo usa dados gravados com uma análise objetiva induzida (e.g., a modelagem de jogador de Houlette (Houlette04)); interpretada, quando a modelagem é mais subjetiva que a induzida, podendo requerir uma intervenção humana para as observações empíricas em descrições informativas; analítica, quando o modelo é criado a partir da utilização métodos automatizados que extraem características do próprio *game design* (e.g., uso de métodos de otimização e de aprendizado); sintética, quando o modelo é derivado de um conceito externo ao próprio jogo (e.g., a definição de fluxo e de diversão citadas no capítulo 2).

4.2

Usos de modelagem de jogador

A modelagem de jogador é tradicionalmente usada no contexto de detecção de preferências de jogador e ajuste dinâmico de dificuldade, como visto no capítulo 3. Machado et al. (Machado11a, Machado11c) identificam também o uso de modelagem de jogador como ferramenta no auxílio ao *game design*, como forma de diversificar e tornar NPCs similares a jogadores reais e em *storytelling* interativo, podendo ser utilizada para melhorar a experiência do interator com a narrativa.

Além dos usos dentro da adaptatividade em um jogo específico, Houlette (Houlette04, p. 565) propõe que o modelo de jogador possa ser exportado para outros jogos que usem uma modelagem compatível. Outro uso possível seria como forma de *feedback* ao jogador, apresentando seu modelo final e evolução em forma de gráficos, porém com a possibilidade de expor a inteligência artificial ao jogador e com isso permitir sua exploração.

4.3

Técnicas

Houlette diz que: “...escolher o conjunto correto de características para seu modelo de jogador é, como muito do desenvolvimento de jogos, mais arte que ciência.” (Houlette04, p. 558), sendo que o modelo é dependente do jogo em si e frequentemente incompatível com outros jogos. Suas propostas de regras gerais para o desenvolvimento de modelos de jogador são: capturar os aspectos importantes da jogabilidade no modelo; modelar de forma a complementar as estratégias da inteligência artificial; ou montar uma modelagem que sirva de base para definir a especificação de requisitos da inteligência artificial. Houlette sugere também a utilização de modelos hierárquicos de jogador (Houlette04, p. 563–564), onde certas características podem ser compostas de outras características mais específicas, como visto na Figura 4.1.

Machado et al. (Machado11a) discutem diversas técnicas usadas para modelagem de jogador: funções de avaliação, redes neurais, modelos baseados em regras, máquinas de estado finito, modelos probabilísticos, modelos baseados em casos e algoritmos evolutivos. Demasi e Cruz (Demasi03c) usam lógica *fuzzy* para construir modelos de jogador para adaptatividade, criando regras a partir do aprendizado com o jogador.

O uso de funções de avaliação envolve uma busca em um espaço de possibilidades, organizado como um grafo (Machado11a), usando a função de avaliação para comparar o modelo atual do jogador.

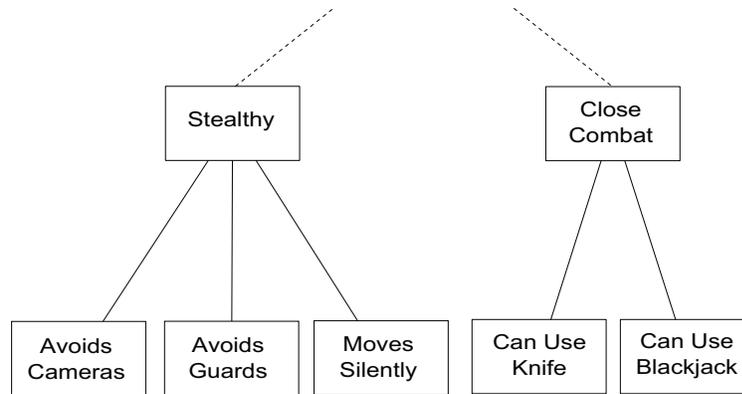


Figura 4.1: Exemplo de modelo de jogador hierárquico. Retirado de (Houlette04, p. 563).

Redes neurais podem ser usadas para coletar estatísticas de diversos jogadores humanos ou agentes inteligentes e a formar modelos de jogadores partir desses dados, usando algoritmos de *clustering*. Os trabalhos de Yannakakis et al. (Yannakakis05, Yannakakis08a, Yannakakis08b, Pedersen09) usaram redes neurais para gerar modelos de jogador e analisam seu impacto na experiência de jogo através da satisfação do jogador.

Modelos baseados em regras são séries de precondições que se satisfeitas resultam em ações. Uma variação desses modelos são máquinas de estado finito, máquinas *fuzzy* de estado finito, modelos probabilísticos e árvores comportamentais.

5 Metodologia

Neste capítulo, o jogo especialmente desenvolvido para os testes é descrito, o algoritmo de jogo adaptativo é proposto e os principais aspectos de implementação são apresentados. Este capítulo também apresenta o método de avaliação da experiência do jogo.

5.1 Tipo do jogo

Para a avaliação do uso de dificuldade adaptativa em jogos, foi desenvolvido um jogo do tipo *shoot'em up* classificado pela taxonomia de Gularte apresentada na seção 2.1.2 como um jogo de tiro de ação *single player*.

Os jogos *shoot'em up* também são conhecidos como *shmups* ou coloquialmente como “jogos de navinha”. Eles constituem um gênero de jogos popular em *arcades* e em especial no Japão, sendo tradicionalmente em *scroll* horizontal ou vertical, tipicamente em 2D. Representantes populares do gênero¹ são Space Invaders, R-Type, Gradius, Ikaruga e Dodonpachi.

A Figura 5.1 ilustra o aspecto do jogo proposto.

5.2 Algoritmo para jogo adaptativo

A modelagem do grau de dificuldade do jogo proposto e o processo de ajuste do jogo são baseados na hipótese de que o comportamento do NPC é determinado por um conjunto V de variáveis, chamadas de variáveis de comportamento do NPC. No jogo *shoot'em up* proposto, temos:

$$V = \{speed, shotDelay, halfRange\} \quad (5-1)$$

¹Para mais informações sobre o gênero, recomendamos como leitura: <http://www.racketboy.com/retro/shooters/shmups-101-a-beginners-guide-to-2d-shooters> e <http://www.racketboy.com/retro/shooters/games-that-defined-the-shmups-genre>.



Figura 5.1: Exemplo de situação de jogo

Os valores destas variáveis nascem iguais aos seus valores de referência:

$$speed = speed_0 \quad (5-2a)$$

$$shotDelay = shotDelay_0 \quad (5-2b)$$

$$halfRange = halfRange_0 \quad (5-2c)$$

Os valores de referência são estabelecidos pelo programador. No jogo em questão, os valores escolhidos foram: $speed_0 = 300$, $shotDelay_0 = 900$ e $halfRange_0 = 200$.

Estas variáveis são a base para os comportamentos de mais alto nível de um NPC, tais como agilidade e precisão. A variável $speed$ tem efeito direto no cálculo de deslocamento do NPC, de maneira que quanto mais alto for esse valor, mais experiente o jogador terá que ser para enfrentar este NPC mais ágil. O atraso entre cada tiro ($shotDelay$) afeta a taxa de tiro ($rate\ of\ fire$) do NPC e, portanto, um valor baixo desta variável significa um NPC mais eficiente. A variável $halfRange^2$ determina a área de ameaça, que estabelece a precisão do tiro do NPC. Quanto menor for esta área, mais preciso será o tiro

² $halfRange$ tem interpretações diferentes dependendo do tipo do jogo. Por exemplo, em jogos FPS (*First-Person Shooter*) $halfRange$ está associada ao decaimento de dano (*damage falloff*) de uma dada arma que é geralmente governado por duas variáveis: máximo alcance ($maxRange$) e percentagem mínima de dano (p). Neste caso, a arma causa 100% de dano para os primeiros $p \times maxRange$ e a partir deste ponto cai linearmente para 0% de dano ao longo da trajetória até alcançar a distância igual a $maxRange$. Conseqüentemente, o dano em $halfRange = \frac{maxRange}{2}$ é $\frac{50}{(100-p)} \approx 56\%$.

e, conseqüentemente, mais experiente deverá ser o jogador para enfrentar este NPC mais letal.

Uma maneira de ajustar o jogo para um determinado grau de dificuldade é criar um fator multiplicador (associado a cada tipo de jogador) que afeta as variáveis de comportamento. No nosso jogo, consideramos o seguinte conjunto de tipos de jogador:

$$\text{Tipos} = \text{easy}, \text{medium}, \text{hard} \quad (5-3)$$

e definimos o multiplicador de dificuldade $m(\text{tipo})$, que é neutro (i.e. igual a 1.0) para o tipo *medium*. No nosso jogo, consideramos o seguinte:

$$m(\text{easy}) = 0.85 \quad (5-4a)$$

$$m(\text{medium}) = 1.0 \quad (5-4b)$$

$$m(\text{hard}) = 1.2 \quad (5-4c)$$

Com base nestes conceitos, o ajuste do tipo (ou modelo³) de jogador é feito aplicando o multiplicador sobre as variáveis de comportamento, como no Algoritmo 5.2. Exploramos essa aplicação em detalhes na seção 5.3.4.

Algoritmo 5.1 function ajusta(tipo) return valores de V

```

speed ← speed0 × m(tipo)
shotDelay ←  $\frac{\text{shotDelay}_0}{m(\text{tipo})}$ 
halfRange ←  $\frac{\text{halfRange}_0}{m(\text{tipo})}$ 
newV ← {speed, shotDealy, halfRange}
return newV

```

A remodelagem do tipo de jogador é baseada na definição de um conjunto C de n características de desempenho do jogador, também chamadas de *traits*, tais que:

$$c_i \in C / c_i \in [0, 1], i = 1, n \quad (5-5)$$

E cujos valores são calculados no final de cada *wave*⁴ de inimigos. A remodelagem de jogador implementada é explicada em detalhes na seção 5.5.

A relação entre as características de desempenho do jogador e o tipo de jogador é dada por valores mínimos e máximos de desempenho padrão

³Neste trabalho, usamos livremente os termos “tipo de jogador” e “modelo de jogador” como sinônimos, apesar de que, rigorosamente falando, “modelo de jogador” e “modelagem de jogador” referem-se a todo o conjunto da proposta.

⁴*Wave* é o conjunto de inimigos que surgem juntos. É uma terminologia típica de jogos de tiro com formação de naves inimigas (e.g. Space Invaders).

associados a cada tipo: $c_{i,min}^{tipo}$ e $c_{i,max}^{tipo}$. Por exemplo, $c_{2,min}^{medium} = 0.3$ e $c_{2,max}^{medium} = 0.6$, conforme ilustra a Tabela 5.5.1.

Desta maneira, definimos os valores mínimos e máximos que identificam um determinado tipo de jogador:

$$MIN^{tipo} = \sum_{i=1}^n c_{i,min}^{tipo} \quad (5-6a)$$

$$MAX^{tipo} = \sum_{i=1}^n c_{i,max}^{tipo} \quad (5-6b)$$

A Tabela 5.5.1 mostra, por exemplo, que $MIN^{medium} = 1.2$ e $MAX^{medium} = 2.4$.

Considerando todas as definições acima, podemos apresentar o nosso algoritmo de jogo adaptativo (Algoritmo 5.2), que é baseado no framework de Charles e Black (Charles04) e no método de adaptação de jogo proposto por (Houlette04):

Algoritmo 5.2 Algoritmo de jogo adaptativo

```

 $\alpha \leftarrow \text{learningRate}$ 
 $tipo_0 \leftarrow$  tipo inicial informado pelo jogador
 $c_i \leftarrow \frac{(c_{i,min}^{tipo_0} + c_{i,max}^{tipo_0})}{2}$  {i.e., a média do desempenho padrão do  $tipo_0$  para a
característica  $c_i$ .}
 $V \leftarrow$  conjunto inicial de valores de variáveis de comportamento
for all waves do
   $c_{i,obs}$  é o valor observado da característica  $i$ 
   $c_i \leftarrow c_i + \alpha \times (c_{i,obs} + c_i)$  {i.e., atualiza cada característica pela regra de
LMS.}
   $desempenho \leftarrow \sum_{i=1}^n c_i$ 
  if  $desempenho \in [MIN^{tipo}, MAX^{tipo}]$  then
     $novoModelo \leftarrow tipo$ 
  else if  $modeloCorrente \neq novoModelo$  then
    Remodela jogador:
     $V \leftarrow ajusta(modeloCorrente)$ 
  else
    Mantém modelo atual
  end if
  Armazena estatística da wave
end for

```

5.3

Definição do jogo

A linguagem escolhida para o desenvolvimento foi C++ usando Lua⁵(Ierusalimschy06) como linguagem de script e o motor de jogos gratuito e de código aberto ClanLib⁶. Para a arte e som, foram usados *assets* livres, sob licença GPL⁷ ou sob licença Creative Commons⁸, como os gráficos disponibilizados na biblioteca SpriteLib⁹, gráficos e efeitos sonoros do site OpenGameArt¹⁰ e as músicas no site Jamendo¹¹.

De acordo com (Bernhaupt10, p. 5–6) e (Novak11, p. 340–352), o jogo desenvolvido pode ser considerado como estando na fase alfa de desenvolvimento, onde testes de jogabilidade são executados para realizar a avaliação do sistema desenvolvido. No caso, nossa avaliação apresentada no capítulo 6, além de estar relacionada com a corretude do sistema adaptativo desenvolvido, está relacionada também à percepção e aceitação do sistema pelos jogadores.

5.3.1

Seleção de dificuldade

Duas versões do jogo foram desenvolvidas neste trabalho: uma com adaptatividade e outra sem. Ambas as versões do jogo contam com a seleção de dificuldade através de um menu (Figura 5.2). A seleção permite uma calibragem inicial do sistema adaptativo, descrito na seção 5.4 de acordo com a preferência do jogador e para o jogo sem adaptatividade seleciona a dificuldade comum a todas as *waves* de inimigos.

5.3.2

Lua

Usamos Lua como linguagem de definição de dados, permitindo configurar parâmetros sem a necessidade de recompilar o código-fonte do jogo. A listagem 5.1 mostra o *script* Lua usado para configurar as opções de jogo, como o multiplicador de cada nível de dificuldade e demais parâmetros com seus valores conforme utilizados nos testes. Também usamos Lua para descrever a cena de testes, com as formações de naves inimigas por *wave* e seus comportamentos.

```
-- Configuration file for Adaptive Shooter
```

⁵<http://www.lua.org>

⁶<http://clanlib.org>

⁷<http://www.gnu.org/copyleft/gpl.html>

⁸<http://creativecommons.org/>

⁹<http://www.widgetworx.com/widgetworx/portfolio/spritelib.html>

¹⁰<http://opengameart.org/>

¹¹<http://www.jamendo.com/>

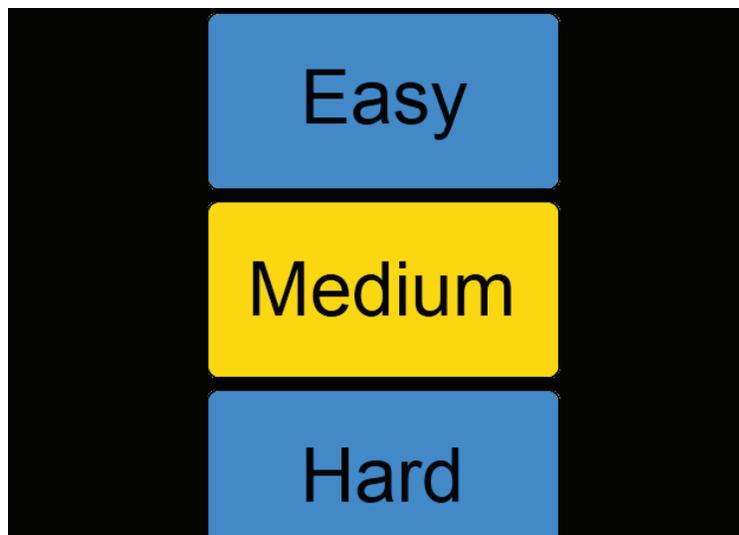


Figura 5.2: Menu de seleção de dificuldade.

```
Player = {  
  SpeedX = 250,  
  SpeedY = 250,  
  Resource = "sprites/rwing",  
  Lives = 5,  
  LearningRate = 0,  
  ShotSpeedX = 0,  
  ShotSpeedY = -500,  
  ShotDelay = 450,  
  HitBoxScale = 0.25  
}
```

```
Enemies = {  
  ShotSpeedX = 0,  
  ShotSpeedY = 300,  
  ShotDelay = 900,  
  EasyMultiplier = 0.85,  
  NormalMultiplier = 1.0,  
  HardMultiplier = 1.2  
}
```

Listagem 5.1: Configurações de teste em config.lua

5.3.3

Esquema de controle

Foi permitido aos jogadores escolherem o esquema de controle que preferissem dentre as implementações por teclado ou *gamepad*. Para os testes, usamos um *gamepad* de Xbox360 por sua compatibilidade com o sistema e motor de jogo utilizados. Para o teclado, foram utilizadas as setas direcionais para movimentar o avatar do jogador pela tela e a tecla Z para disparar os tiros e confirmar a seleção de dificuldade. Para o *gamepad*, foi utilizada a alavanca analógica esquerda (com interpretação digital do valor de resposta) para movimentar o avatar do jogador e o botão X para disparar os tiros e confirmar a seleção de dificuldade.

A oferta de ambos os métodos de controle para os jogadores se justifica por permitir que o jogador escolhesse o que mais lhe agradasse, procurando favorecer as condições para que o jogador entrasse em fluxo, ao invés de obrigar o jogador a usar uma opção de controle que não lhe fosse familiar ou que não lhe agradasse.

5.3.4

Gameplay

O *gameplay* do jogo implementado consiste em o jogador, representado pela nave que controla, desviar dos tiros e das naves inimigas que se movimentam da parte superior da tela em direção à parte inferior, enquanto busca abater a maior parte possível dos inimigos para aumentar sua pontuação. Cada colisão do jogador com um tiro disparado pelos inimigos ou contra uma nave inimiga custa-lhe uma vida. Findas as cinco vidas (dada a configuração apresentada na listagem 5.1), o jogo termina com uma tela de *game over* (Figura 5.3). Caso o jogador sobreviva a todas as *waves* de inimigos, é vitorioso e apresentado à tela de vitória (Figura 5.4).

Foram implementados dois tipos de inimigos: discos voadores, que se movimentam em formação em asa; caças, que se movimentam em zig zag pela tela ou restritos a um corredor imaginário (exemplificados na Figura 5.1). A variação da inteligência artificial desses inimigos é feita através de uma variável de comportamento, com ajustes influenciados pelos multiplicadores de dificuldade apresentados na listagem 5.1 em sua velocidade, atraso entre os tiros e área de ameaça. A velocidade é modificada diretamente pelo multiplicador de dificuldade. A área de ameaça simula uma inteligência para a precisão dos disparos do inimigo, onde quanto menor a dificuldade, maior a área de ameaça (definida pelo intervalo $[SpriteCenter - halfRange, SpriteCenter + halfRange]$ onde $halfRange = \frac{200}{m(tipo)}$ e $SpriteCenter$ é a posição do centro do *sprite* do NPC).



Figura 5.3: Tela de fim de jogo.

Como o atraso entre os tiros também é modificado pela dificuldade (definido como $delay = \frac{ShotDelay}{m(tipo)}$), isso permite uma janela de reação maior (em baixa dificuldade) e menor (em alta dificuldade) para o jogador reagir ao disparo e contra-atacar. $m(tipo)$ foi definido no conjunto de equações 5-4.

5.4 Sistema adaptativo

O sistema adaptativo implementado constitui-se de um agente inteligente, o AIManager, que percebe o ambiente do jogo através dos dados de desempenho do jogador e modifica o ambiente alterando variáveis dos NPCs de forma a buscar um nível de dificuldade adequado à variação de desempenho do jogador. Por nível de dificuldade adequado, queremos dizer o ajuste das variáveis dos NPCs que proporcione um desafio ao jogador sem ser fácil demais (identificado por um baixo ou nenhum consumo dos recursos do jogador, como por exemplo pontos de vida de seu avatar e bombas utilizadas) nem difícil demais (identificado por mortes constantes do avatar do jogador com baixa porcentagem de inimigos destruídos).

Escolhemos não alterar as variáveis relacionadas ao jogador pois o jogador pode perceber as alterações ao sentir as diferenças em aspectos como seu deslocamento, seu poder de fogo e sua defesa. Tais mudanças, se não explícitas ao jogador por mecânicas do jogo passivas (como evolução em níveis) ou ativas (por sua escolha, como coleta de *power ups* ou por treinamento de habilidades), podem causar quebra da imersão por violar o contrato de suspensão de descrença das regras de jogo.



Figura 5.4: Tela de vitória.

5.4.1

Modelagem e implementação

O sistema adaptativo foi modelado como uma implementação do *framework* de Charles e Black (Charles04, Charles05), apresentado na seção 3.5 e desenvolvido como uma biblioteca separada do jogo. A classe AIManager funciona como o gerente dos agentes inteligentes (implementações da classe abstrata AIAgent) que são afetados pelas mudanças percebidas do ambiente (i.e., o estado da modelagem de jogador), permitindo a atualização dos agentes e acesso ao modelo de jogador (implementação da classe abstrata PlayerModel). Cada modelo de jogador é composto por características, representadas pela estrutura Traits. O AIManager e suas classes abstratas correlatas são apresentados na Figura 5.4.1.

A atualização da modelagem de jogador é dependente da implementação do cliente da biblioteca. Apresentamos o pseudocódigo da implementação da comparação entre modelos de jogadores no algoritmo 5.5 na seção 5.5. Após a atualização do modelo de jogador observado pela instância de AIManager, a classe cliente faz a chamada ao método updateAgents. Os agentes inteligentes observados pelo AIManager recebem a chamada de seu método virtual puro updateStats(), implementado pelo cliente. Dessa forma, o cliente decide como a atualização dos agentes inteligentes deve proceder de acordo com suas especificidades. Esse processo é detalhado no algoritmo 5.4.1. O método compare() da classe PlayerModel é detalhado no algoritmo 5.5.

O uso no jogo (cliente da biblioteca AIManager) se dá através de implementações das classes abstratas da biblioteca. A Figura 5.6 mostra em

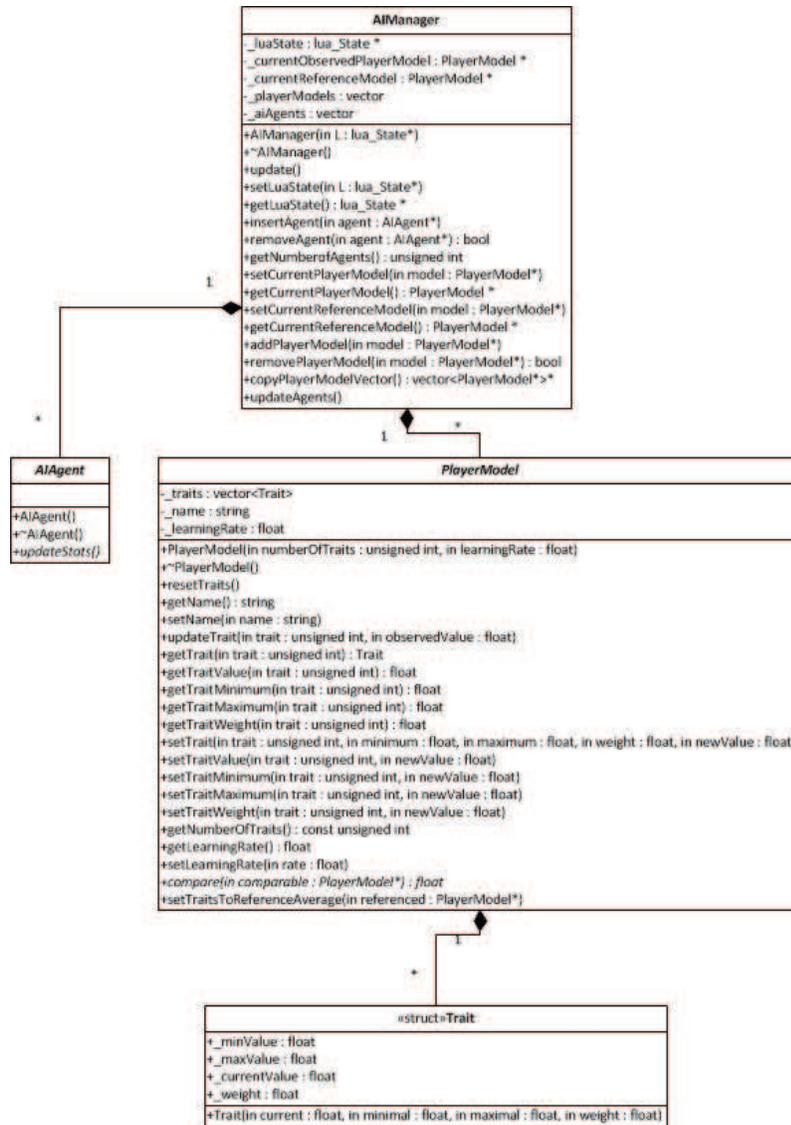


Figura 5.5: Diagrama de classes UML do AIManager

destaque sobre o *framework* de Charles e Black alguns detalhes de nossa implementação através das chamadas de métodos relativos às partes do sistema adaptativo *online*. Cabe notar que o método `waveFinish()` está fora do sistema adaptativo por ser o disparador da chamada do método `update()` do AIManager, já que atualizamos a modelagem de jogador apenas ao final de cada *wave* de inimigos.

O método `updateAgents()` no algoritmo 5.4.1 chama o método virtual `updateStats()` de cada agente inteligente controlado pelo AIManager. Em nossa implementação deste método, verificamos qual a dificuldade do modelo de jogador observado ao final de cada *wave* de inimigos e ajustamos os parâmetros do multiplicador de velocidade, da área de ameaça e do atraso

entre os tiros de cada agente inteligente. O fator de aprendizado α com valor de 0.3 (ver algoritmo 5.5 e equação 5-7), conforme observado por Houlette (Houlette04, p. 560), permitiu uma avaliação com atualização efetiva do modelo de jogador para a dificuldade correspondente aproximadamente a cada três *waves* de inimigos.

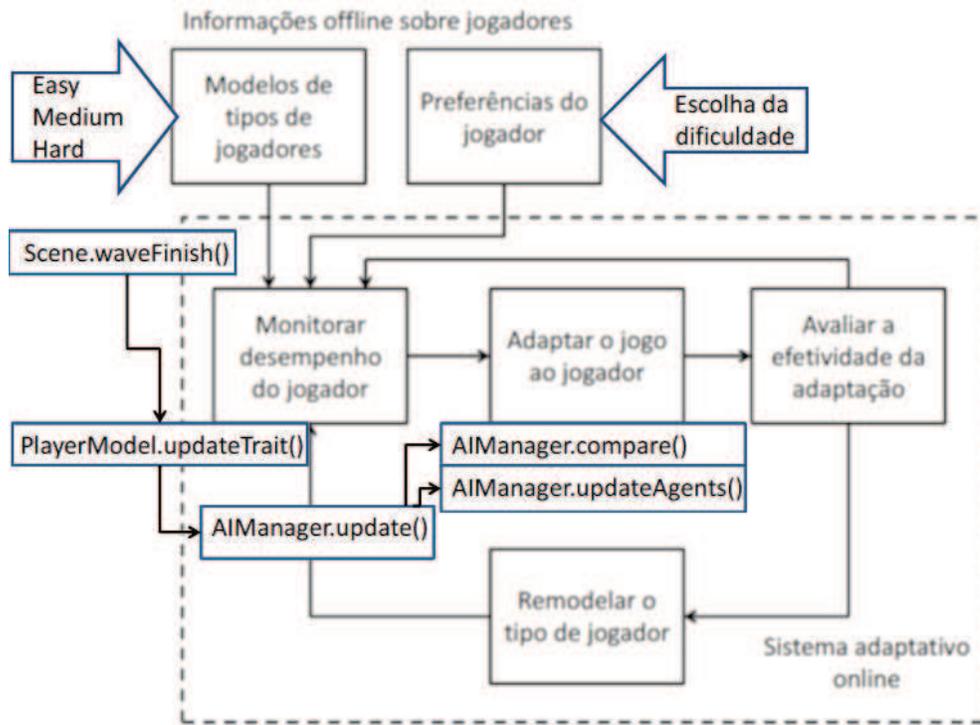


Figura 5.6: Sobreposição de partes da implementação ao *framework* de Charles e Black. Adaptado de (Charles04).

5.5 Implementação da modelagem do jogador

Para a modelagem do jogador, decidimos nos basear na modelagem proposta por Houlette (Houlette04, p.559–560). No modelo de Houlette, cada característica observada encontra-se no intervalo real de 0 a 1, onde zero significa ausência ou “o jogador nunca faz isso” e um significa presença total ou “o jogador sempre faz isso”. Assim, a característica é definida como:

$$\forall x \in X / x \in \mathbb{R}, 0 \leq x \leq 1$$

Onde:

X = Conjunto de características do modelo de jogador.

As características observadas foram:

Algoritmo 5.3 function AIManager.update()

```

result ← 0
for playerModelIterator ← playerModels.begin() to playerModels.end()
do
  result ← currentObservedModel.(playerModelIterator)
  if result < 0 then
    continue
  else if result = 0 then
    currentReferenceModel ← playerModelIterator
  else
    result ← currentReferenceModel.compare(playerModelIterator)
    if result < 0 then
      currentReferenceModel ← playerModelIterator
    end if
  end if
end for
currentObservedModel.setName(currentReferenceModel.getName())
updateAgents()

```

Taxa de acerto: Definida pelo número de tiros disparados pelo jogador pelo número de tiros acertados em naves inimigas. Indica a precisão dos disparos do jogador, onde um valor 1 indica que todos os disparos foram bem sucedidos e um valor 0 indica que nenhum disparo acertou as naves inimigas.

Variação de vidas: Taxa de variação de vidas do jogador. Definido pelo número de vidas restantes ao final da *wave* pelo número no início desta. Um valor 1 indica que o jogador perdeu todas as vidas e um valor 0 indica que nenhuma vida foi perdida.

Taxa de inimigos destruídos: Definida pelo número de inimigos abatidos até o final da *wave* pelo número total de inimigos da *wave*. Um valor 1 indica que todos os inimigos foram destruídos, enquanto um valor 0 indica que nenhum inimigo foi destruído.

Taxa de inimigos destruídos total: Definida pelo número total de inimigos abatidos pelo número total de inimigos no jogo. Um valor de 1 indica que todos os inimigos foram destruídos, enquanto um valor 0 indica que nenhum inimigo foi destruído.

Para a atualização do valor das características ao final de cada *wave* de inimigos, como sugerido por Houlette (Houlette04, p. 560), usamos um método baseado na média dos mínimos quadrados (LMS, *Least Mean Squares*), como mostrado na equação 5-7.

$$\text{traitValue} = \alpha \cdot \text{observedValue} + (1 - \alpha) \cdot \text{traitValue} \quad (5-7)$$

Onde:

α = Constante de taxa de aprendizado.

Esta equação é inspirada na regra de aprendizagem supervisionada conhecida por “regra de atualização LMS” (*LMS update rule*), por “regra de aprendizado de Widrow-Hoff” ou por “regra delta” (*delta rule*). Widrow e Hoff (Widrow88, pp. 123–134) treinaram redes neurais simples usando esta regra especialmente voltada para aprendizagem *online*.

Através do algoritmo implementado aplicado a cada uma das características observadas (ver algoritmo 5.5), o modelo de jogador em uma determinada seção do jogo é uma suposição do AIManager e cada chamada à função de atualização é um refino deste modelo através de uma nova evidência (o novo valor observado). A influência da função sobre a atualização da característica é limitada por uma constante de taxa de aprendizado entre 0 e 1 definida empiricamente. A necessidade de se limitar o ajuste é dada pela incerteza do jogador agir de forma completamente consistente entre cada período de atualização, o que pode levar a grandes flutuações no modelo, segundo Houlette.

Algoritmo 5.4 function Player.updateTrait(trait, observedValue)

```

currentValue ← traits[trait].currentValue
δ ← observedValue – currentValue
weightedδ ← learningRate × δ
setTraitValue(trait, traits[trait].currentValue + weightedδ)

```

Após a atualização das características do modelo de jogador, é necessário verificar se o modelo atual ainda se encontra dentro do modelo anteriormente atribuído ou se pertence a outro modelo. A comparação das características do modelo atual com o modelo atribuído indica se a atualização do modelo é necessária e é feita como no algoritmo 5.5 e pode ser visto como um *perceptron* de uma rede neural, embora neste caso tenhamos apenas um neurônio.

5.5.1

Modelos de jogador implementados

Para o jogo, implementamos modelos de jogadores baseados em três tipos de dificuldade: *Easy*, *Medium* e *Hard*. Os modelos foram construídos baseados nas quatro características observadas citadas anteriormente, taxa de acerto, variação de vidas, taxa de inimigos destruídos por *wave* e taxa de inimigos destruídos total. Determinamos empiricamente os valores para cada modelo de jogador, apresentados na Tabela 5.5.1.

Algoritmo 5.5 function PlayerModel.compare(comparable)

```

total ← 0
totalMin ← 0
totalMax ← 0
for i ← 0 to numOfTraits do
    total ← total + getTraitValue(i)
    totalMin ← totalMin + comparable.getTraitMinimum(i)
    totalMax ← totalMax + comparable.getTraitMaximum(i)
end for
if total < totalMin then
    return -1
else if total > totalMax then
    return 1
else
    return 0
end if

```

Tabela 5.1: Modelos de jogador implementados

	Easy		Medium		Hard	
	Min	Max	Min	Max	Min	Max
Taxa de acertos	0.0	0.3	0.3	0.6	0.6	1.0
Varição de vida	0.6	1.0	0.3	0.6	0.0	0.3
Inimigos por wave	0.0	0.3	0.3	0.6	0.6	1.0
Inimigos total	0.0	0.3	0.3	0.6	0.6	1.0
Totais	0.6	1.9	1.2	2.4	1.8	3.3

Para tais intervalos de mínimo e máximo totais dos modelos de jogador, percebe-se que há intercessões entre os pares de modelos *Easy-Medium* e *Medium-Hard*, com valor de 0.6. Como os modelos de jogador estão ordenados no vetor de comparação, essa intercessão permite uma ascensão mais lenta nos níveis de dificuldade e uma queda mais rápida. O objetivo é procurar reduzir a frustração do jogador com uma dificuldade elevada.

5.6**Testes com jogadores**

A literatura apresenta diversas formas de se realizar testes com jogadores para detectar diversão e a experiência de fluxo, dentre as quais podemos citar (Lankes10), (Koeffel10), (Schell11, p. 122). Nacke e Lindley (Nacke10) fazem uso de eletromiografia facial para analisar a resposta de músculos faciais de modo a detectar variações de emoção no jogador. Questionários pós-teste são comumente utilizados para avaliação da experiência de jogo e envolvimento do jogador (Brockmyer09, Calvillo-Gámez10, Ibáñez11, Machado11b).

Para poder avaliar o uso do sistema adaptativo desenvolvido, foram

realizados testes com jogadores. Os testes realizados neste trabalho envolveram os jogadores jogarem ambas as versões desenvolvidas (com e sem dificuldade adaptativa) em ordem aleatória e responderem o questionário de avaliação da experiência de jogo. Neste capítulo, apresentamos os conceitos abordados nos testes com os jogadores e justificamos as escolhas na metodologia de testes. Os questionários encontram-se no apêndice A.

Cabe ressaltar que nesta seção nos referimos a usuário, participante, testador e jogador como sinônimos dentro do ambiente de teste, não distinguindo entre cada nome utilizado.

5.6.1

User experience em jogos

A definição de *user experience* segundo o ISO é: “as percepções e respostas de uma pessoa que resultam do uso ou uso antecipado de um produto, sistema ou serviço” (Bernhaupt10, p. 4, tradução nossa). Em jogos, trata-se do relacionamento do jogador com o jogo enquanto este acontece, tanto o processo quanto o resultado do jogar.

Calvillo-Gómez et al. (Calvillo-Gómez10, p. 50) mencionam que é possível avaliar jogos em relação à experiência, embora seja subjetiva, por serem compostos por três propriedades que permitem o foco na atividade: objetivo, ferramenta e domínio. O objetivo é o que se pretende alcançar com a atividade. As ferramentas são os meios pelos quais se atingirá o objetivo e o domínio está relacionado com o contexto da atividade. Por exemplo, em um jogo, o objetivo poderia ser a catarse (o jogador deseja uma experiência catártica, em que há divergência do seu cotidiano, um escape), a ferramenta é o jogo que ele utiliza para atingir tal objetivo e o domínio é o contexto do jogo (o seu gênero e ambientação).

Calvillo-Gómez et al. (Calvillo-Gómez10, p. 50) identificam três qualidades necessárias à aplicação para que o usuário tenha foco na atividade: a ferramenta precisa ser funcional, isto é, ser capaz de realizar a tarefa a que se destina; a ferramenta precisa ser usável, isto é, suas propriedades são compatíveis com as necessidades do usuário, relacionamento realizado através de conceitos como eficiência, eficácia e *affordance*¹²; e esteticamente atraente, de forma que dentre várias ferramentas de uso e função semelhante, o fator que leve o usuário a selecionar aquela.

Através dessas propriedades e qualidades, pode-se avaliar jogos similares e diferentes em relação à experiência de jogo proporcionada ao jogador, seja

¹²O conceito de *affordance* pode ser entendido como a relação entre as ações possíveis de se realizar com a ferramenta e as ações possíveis percebidas.

através de entrevista com o jogador usando questionários pós-jogo, seja através de avaliações usando heurísticas realizadas por peritos na área. Na seção 5.6.2 vemos como avaliar a experiência de jogo de um jogador usando um questionário pós-jogo.

5.6.2

Avaliação da experiência de jogo

Como metodologia de avaliação da experiência de jogo, optamos pelo uso de questionários em momento pré-teste, para poder classificar o usuário conforme seu conhecimento anterior em jogos, e pós-teste para avaliar os impactos da experiência de jogo.

Quanto ao uso da estratégia de *think aloud* (Fullerton08, p. 264) para avaliação da experiência, optamos pela alternativa descrita por Hoonhout (Hoonhout08, p. 71), anotando as reações do jogador e realizamos uma entrevista pós-teste, procurando usar as informações coletadas durante o teste como guia para perguntas adicionais. O uso da alternativa de Hoonhout visa evitar a quebra da imersão que pode ser provocada ao instruir o participante a verbalizar sua experiência.

Os testes com o jogo desenvolvido foram conduzidos de acordo com as recomendações de *playtesting* de Fullerton (Fullerton08, pp. 252–269). Esta subseção detalha e justifica o uso das metodologias anteriormente citadas.

O framework CEGE

O *framework* CEGE (*Core Elements of the Game Experience*) proposto por (Calvillo-Gómez10) parte do entendimento de que a experiência de jogar envolve o videogame em si e a interação entre o jogador e o videogame. Esta interação, chamada de “puppetry”¹³ por Calvillo-Gómez et al. (op. cit.), provê a ligação para o resultado da experiência. A metáfora da manipulação de marionetes (*puppetry*, ou titerismo) serve para descrever esta experiência de jogar, ou, mais precisamente, serve para descrever a parte interativa da experiência. *Puppetry*, de acordo com (Calvillo-Gómez10), descreve como o jogador começa se aproximando do videogame até que, eventualmente, o jogo sendo jogado é o resultado das ações do jogador.

Procurando contribuir para um melhor entendimento do *framework* de Calvillo-Gómez et al. (op. cit.), a presente dissertação propõe que os elementos

¹³*Puppetry* significa a produção, a criação ou a manipulação de *puppets* (marionetes). A tradução mais referenciada é “fantocheda”, o que, entretanto, não reflete o conceito. Optamos pela tradução titerismo.

centrais (*core elements*) de Calvillo-Gómez et al. (op. cit.) sejam apresentados conforme a organização de sua hierarquia:

- Elementos Principais da experiência de Jogar (i.e., *Puppetry* e *Video-game*)
 - Elementos Construtores que permitem a percepção dos Elementos Principais
 - * Elementos Observáveis do Processo que são consequências dos Elementos Construtores

A Figura 5.7 apresenta esta hierarquia, junto como a indicação das variáveis latentes e observáveis propostas por (Calvillo-Gómez10). Os questionários para os jogadores são criados usando as variáveis observáveis (que são elementos que podem ser medidos diretamente).

Variáveis Latentes	Puppetry			Videogame		Elementos Principais
	Control	Ownership	Facilitators	Game-Play	Environment	Elementos Construtores
	Small Actions	Big Actions	Time	Rules	Graphics	
Variáveis Observáveis	Controllers	Personal Goal	Aesthetic Value	Scenario	Sound	Elementos Observáveis
	Memory	You but not You	Previous Experiences			
	Point of View	Rewards				
	Goal					
	Something to do					

Figura 5.7: Elementos centrais da Experiência de Jogo. Adaptado de (Calvillo-Gómez10)

Os elementos centrais de Calvillo-Gómez et al. (op. cit.) também têm uma estrutura de relacionamento. O principal relacionamento é a constatação de que a experiência positiva, (i.e., *enjoyment* (diversão)), é alcançada pela percepção do *videogame* pelo jogador e pela interação dele com o *videogame* (*puppetry*). Isto é, os elementos *puppetry* e *videogame* produzem diversão, assim como a ausência deles provoca a experiência negativa oposta: a frustração. *Enjoyment* (diversão) é considerada uma variável latente do *framework*, junto com os outros 5 elementos construtores. Aliás, outras variáveis latentes podem ser consideradas: “frustração”, “*puppetry*”, “*videogame*” e a própria experiência CEGE.

O outro relacionamento vem da constatação de que Controle (*Control*) e Facilitadores (*Facilitators*) produzem Posse (*Ownership*). Na próxima seção, as

descrições dos Elementos Construtores (variáveis latentes) e de outras variáveis são apresentadas. A hierarquia do *framework* (Figura 5.7) e a estrutura de relacionamento entre as variáveis estão graficamente apresentadas na Figura 5.8. A descrição das variáveis observáveis e a explicação de como elas são usadas para criar o questionário podem ser encontradas na tese de doutorado de Eduardo Calvillo-Gámez (Calvillo-Gámez09).

Questionário pré-teste

Em Calvillo-Gámez et. al (Calvillo-Gámez10, p. 50) temos que experiências pretéritas afetam experiências futuras. Desse modo, aplicamos um questionário anterior ao teste para obter informações qualitativas como idade, sexo, e se o jogador já teve alguma experiência com jogos antes do teste e como ele se classificaria em nível de habilidade com jogos, dividindo entre casual e dedicado (i.e., *hardcore*). Essa diferenciação foi escolhida para auxiliar na compreensão dos resultados, baseado nos critérios de diferenciação apresentados na seção 2.2.1.

A expectativa é que jogadores dedicados tenham uma facilidade maior em assimilar a experiência de jogo que jogadores casuais. Essa classificação entre casual e dedicado é reforçada pelas perguntas sobre quantas horas o jogador dedica para jogos na semana e com quais gêneros está familiarizado. Também esperamos que os jogadores que se declararam como casuais escolham a dificuldade inicial como fácil (*easy*) ou normal (*medium*) e que os jogadores que se declararam como dedicados escolham normal (*medium*) ou difícil (*hard*).

Além desses dados, coletamos também informações sobre a média semanal de horas gastas jogando jogos de *videogame* e quais gêneros de jogos está familiarizado. Os gêneros foram adaptados dos supergêneros identificados no relatório de 2012 da ESA (ESA12) e dos gêneros identificados no Australian Centre For the Moving Image¹⁴ por se aproximarem mais das classificações usadas em outras mídias com as quais os jogadores estão mais familiarizados, como revistas e *sites* especializados, do que a taxonomia de Gularte apresentada na Tabela 4.1. Tais dados servem como apoio à confirmação da autoclassificação do jogador e esperamos que jogadores familiarizados com os gêneros Shoot'em Up e Arcade tenham maior desempenho e facilidade em assimilar os conceitos do jogo.

Questionário pós-teste

Para avaliarmos a diferença de experiência do jogador com o jogo desenvolvido em ambas as versões, usamos o questionário de Elementos Centrais da

¹⁴http://www.acmi.net.au/explore_game_genres.htm

Experiência de Jogo (CEGE) de Calvillo-Gómez et al. (Calvillo-Gómez10), que se baseia no *framework* CEGE apresentado anteriormente nesta dissertação, usando uma escala Likert de sete pontos para as questões. O questionário é aplicado posteriormente à experiência de jogo com cada uma das versões e é usado para acessar os elementos centrais do modelo CEGE.

Para avaliar o questionário pós-jogo, usamos dois conjuntos de fatores¹⁵ do modelo de Calvillo-Gómez et al., a saber:

Diversão: A sensação de diversão ou prazer ao jogar.

Frustração: A sensação de frustração ao jogar.

ECEJ: Elementos centrais da experiência de jogo. Refere-se a titerismo¹⁶ e video-game conjuntamente.

Titerismo: Sensação de controle e domínio do jogo.

Video-game: Refere-se à jogabilidade e ao ambiente, definido pelos elementos gráficos e sonoros do jogo.

Neste conjunto de fatores, titerismo e *video-game* estão correlacionados com diversão. Se há presença de elementos centrais da experiência de jogo, então a frustração deve ser baixa e não correlacionada (Calvillo-Gómez10, p. 65) (a frustração surge da ausência ou mau funcionamento dos elementos centrais da experiência de jogo), mas não há garantia de que diversão será positiva. Identificando a presença de ECEJ, titerismo e video-game são individualmente avaliados.

Controle: Refere-se à sensação de domínio do jogador sobre as ações que o jogo oferece, fazendo o jogo responder às suas ações, manipulando-o.

Facilitadores: Elementos subjetivos da experiência de jogo, como tempo que o jogadores está disposto a gastar com o jogo, valores estéticos do jogo e experiências anteriores com jogos similares.

Posse: Sensação quando o jogador toma as ações do jogo como suas extensões, como resultado de sua ação consciente sobre o jogo.

Ambiente: A apresentação do jogo, gráficos e sons.

Game-play: Refere-se às regras e ao cenário do jogo.

¹⁵Nesta dissertação, usamos livremente os termos “variável latente” e “fator” como sinônimos.

¹⁶Tradução nossa para *puppetry*.

Este conjunto de variáveis busca avaliar a percepção do jogo formada pelo jogador pelas variáveis ambiente e *game-play* que produzem diversão (do conjunto de variáveis anteriormente apresentado). Controle gera posse, que gera diversão. Posse também é produzida por facilitadores para compensar a baixa sensação de controle, através da subjetividade de fatores como experiências anteriores com jogos similares, apreciação estética do jogo e o tempo que o jogador está disposto a gastar com o jogo.

A Tabela 5.2 mostra a quais variáveis cada questão está relacionada. As questões do questionário pós-jogo estão na seção A.2. A Figura 5.8 mostra a relação entre as variáveis do questionário e os elementos centrais da experiência de jogo.

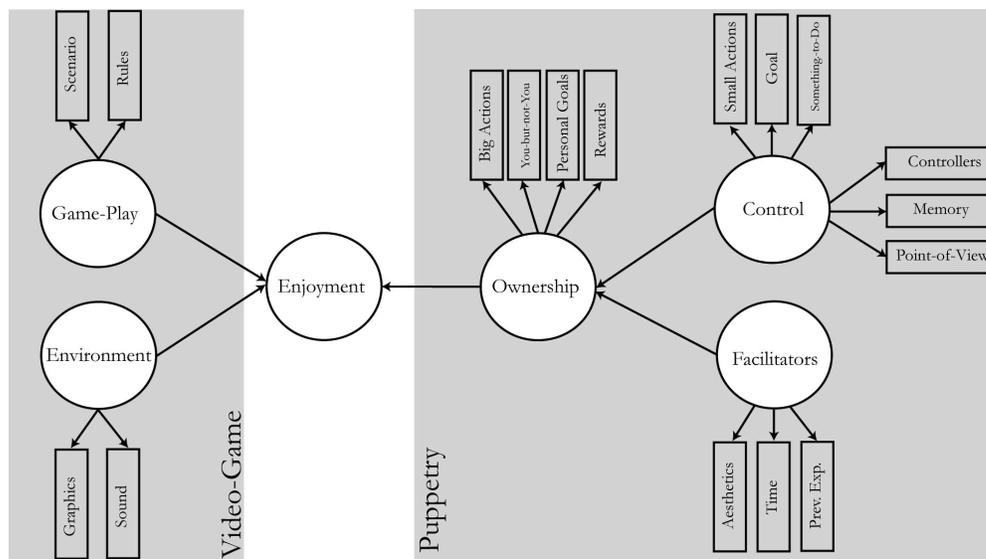


Figura 5.8: Relacionamento entre as variáveis do questionário CEGE, onde os círculos são as variáveis latentes e os retângulos as variáveis concretas e observáveis. Retirado de (Calvillo-Gámez10, p. 64).

Reconhecemos também a existência de outros questionários de avaliação da experiência de jogo, como o GEQ proposto por Brockmyer et al. (Brockmyer09) do Game Experience Lab da Eindhoven University of Technology e a proposta de Parnell (Parnell09) da London University. Entretanto, o modelo CEGE é o que melhor reflete o fato de que a experiência de interagir com um jogo é um fenômeno dual, no sentido de que é ao mesmo tempo um processo e um resultado.

Think aloud

Embora o método de *think aloud* tenha suas vantagens, como permitir compreender o que se passa na cabeça do jogador durante a experiência

Tabela 5.2: Relação das perguntas do questionário com os fatores da experiência de jogo, adaptada de (Calvillo-Gámez10, p. 65)

Itens	Fator
1, 4, 5	Diversão
2, 3	Frustração
6-38	Elementos Centrais da Experiência de Jogo
6-25, 38	Titerismo
26-37	Videogame
6-12, 25, 28	Controle
13-18	Facilitadores
19-25	Posse
26-31	Ambiente
32-37	Game-play

(*concurrent think aloud*), permitindo ao pesquisador inferir sobre o processo cognitivo do usuário a partir dos resultados observados e de suas decisões, sua maior desvantagem é interferir na experiência ao tirar o usuário do contexto da imersão para o contexto do teste, além da verbalização do usuário poder ser influenciada pelo contexto de teste (i.e., pode não se expressar naturalmente, como seria desejado) (Hoonhout08, p. 66-69). Uma alternativa é o uso de filmagem e pedir ao usuário para ver o vídeo e comentar o que pensou naquele momento (*retrospective think aloud*).

Usamos a alternativa descrita por Hoonhout em (Hoonhout08, p. 71), onde o usuário não é instruído para verbalizar sua experiência, mas seus comentários e reações são registrados. Hoonhout identifica que tal método precisa ser amparado por outros métodos de coleta de dados, como fazemos com o uso de entrevista e questionários.

Playtesting

Cada jogador jogou de forma alternada as versões com e sem dificuldade adaptativa, sem saber qual versão estava jogando. A alternância entre os jogadores de qual versão seria jogada primeiro deve-se à necessidade de se eliminar o viés no desempenho causado pelo aprendizado ao jogar uma das versões. Coletamos dados sobre o desempenho de cada jogador durante o jogo de modo a identificar e comparar o nível de dificuldade inicialmente escolhido e o nível de dificuldade percebido pela inteligência artificial, seja este ajustado pelo desempenho (na versão adaptativa) ou mantido (na versão adaptável).

Entrevista

Segundo Hoonhout (Hoonhout08, p. 72), perguntar ao usuário o que ele achou da experiência é o início e o objetivo da entrevista pós-teste.

Através da entrevista, é possível coletar dados como a opinião do participante, comparações com experiências anteriores, pensamentos, atitudes e ideias, complementando dados coletados através de outros meios durante o teste, como observação do protocolo verbal (*think aloud*) e *log* do sistema testado.

Para nossa entrevista com os participantes, escolhemos o formato semi-estruturado, onde predefinimos tópicos para discussão, mas a forma como a pergunta é feita ao participante e a ordem das perguntas é deixada em aberto, de modo a focar em determinados aspectos relativos a cada participante que foram observados durante o teste. Os tópicos encontram-se listados no apêndice A. Tomamos precauções na formulação das perguntas para evitar influenciar os participantes a determinados comportamentos ou respostas, visto que, por ser uma situação social, a entrevista pode ser afetada pelo desejo inconsciente do participante em agradar o entrevistador (Hoonhout08, p. 73).

5.6.3 Participantes

Para evitar vieses nos testes, foram selecionados participantes de ambos os sexos, de diferentes experiências anteriores com jogos e de variadas formações, entre funcionários do laboratórios TeCGraf e ICAD/Visionlab da PUC-Rio, alunos de graduação e pós-graduação dos departamentos de Informática e Artes & Design da PUC-Rio e do departamento de Informática da UFF. Ao todo, a população de testes foi de 35 participantes.

5.6.4 Roteiro dos testes

A aplicação dos testes com os usuários seguiu o seguinte roteiro:

1. Os participantes foram apresentados à pesquisa em que seriam testadores de variações de um jogo. Lembramos aos participantes que não seriam eles o objeto de teste e sim o jogo. Um termo de aceitação foi apresentado aos participantes, que foram instruídos a ler e em caso de aceite, assiná-lo. No termo, os participantes foram avisados de que dados de sua experiência de jogo seriam coletados de forma anônima para que fosse possível uma melhor avaliação do jogo sem identificá-los.
2. Foi aplicado um questionário pré-teste para coletar dados de experiência anterior do jogador.
3. Escolhido de forma alternada, cada participante jogou cada uma das versões do jogo (com ou sem adaptatividade) até que chegasse ao final

do mesmo ou que perdesse todas as vidas. Limitamos a uma única vez por versão.

4. O questionário pós-teste foi aplicado para avaliar a experiência de jogo do participante com a versão jogada.
5. Cada participante jogou a outra versão do jogo, relativa à versão anterior, até que chegasse ao final do demo ou que perdesse todas as vidas.
6. O questionário pós-teste foi aplicado para avaliar a experiência de jogo do participante com a versão jogada.
7. Foi realizada uma entrevista para coletar dados qualitativos a fim de avaliar as diferenças entre o sistema adaptativo e o sistema adaptável.

As perguntas realizadas nos questionários pré e pós teste e na entrevista podem ser conferidas no apêndice A.

6

Análise dos resultados

Neste capítulo apresentamos os resultados dos questionários pré e pós-teste. Quando necessário para justificar a análise, fazemos uso dos *logs* de teste de cada jogador. Essas informações e demais Tabelas encontram-se no apêndice B.

6.1

Análise da população

Como explicado na seção 5.6.3, foram selecionados trinta e cinco participantes para os testes. O primeiro, denominado *P*, foi escolhido como piloto para a calibração do tempo de testes. Para o piloto, obtivemos um tempo de aproximadamente trinta minutos. Esse tempo variou em torno de vinte minutos para os demais participantes. Os participantes estão relacionados na Figura B.1.

Dentre os participantes, as Figuras 6.1 e 6.2 mostram a divisão dos participantes por sexo e por classe (i.e., casual, dedicado ou não-jogador) e a Figura 6.3 mostra a frequência de participantes por classe de idade. As Tabelas 6.1 e 6.2 resumem essas informações. Dentre os participantes, tivemos maioria feminina e apenas um não-jogador.

Tabela 6.1: Resumo dos participantes - Sexo, Classe

Participantes					
Total	Masculinos	Femininos	Casuais	Dedicados	Não-jogadores
35	16	19	18	16	1
%	46%	54%	51%	46%	3%

6.2

Avaliação do sistema adaptativo

A escolha da taxa de acertos (acurácia) como uma das características a serem observadas não foi favorável. Muitos jogadores mantinham o botão de tiro pressionado enquanto moviam a nave pela tela, mesmo sem inimigos ou sem se preocupar em acertá-los, disparando uma chuva de tiros na esperança

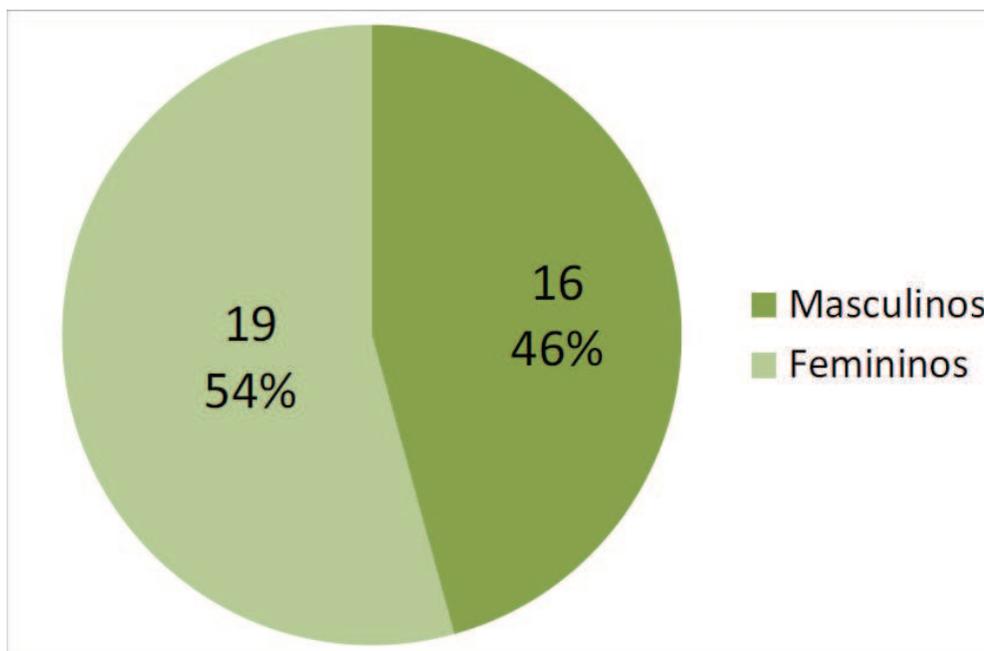


Figura 6.1: Participantes por sexo

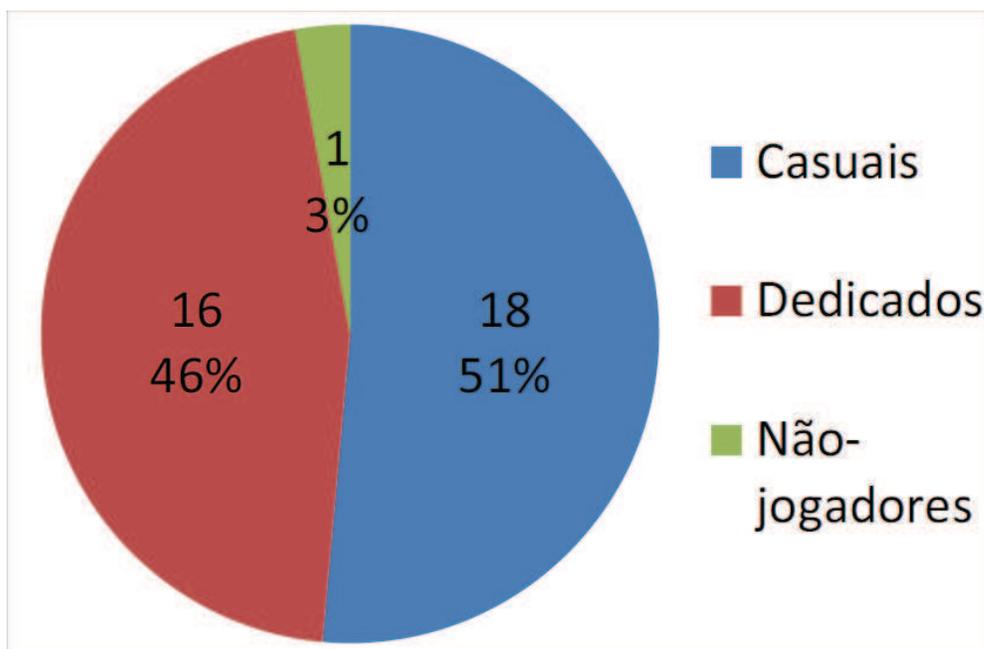


Figura 6.2: Participantes por classe

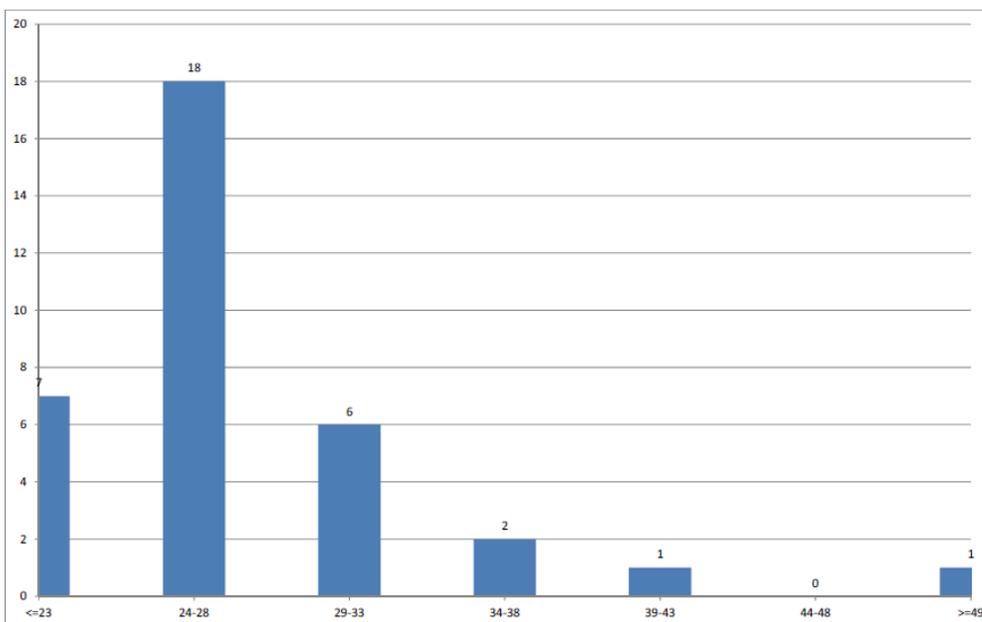


Figura 6.3: Frequência de participantes por classe de idade

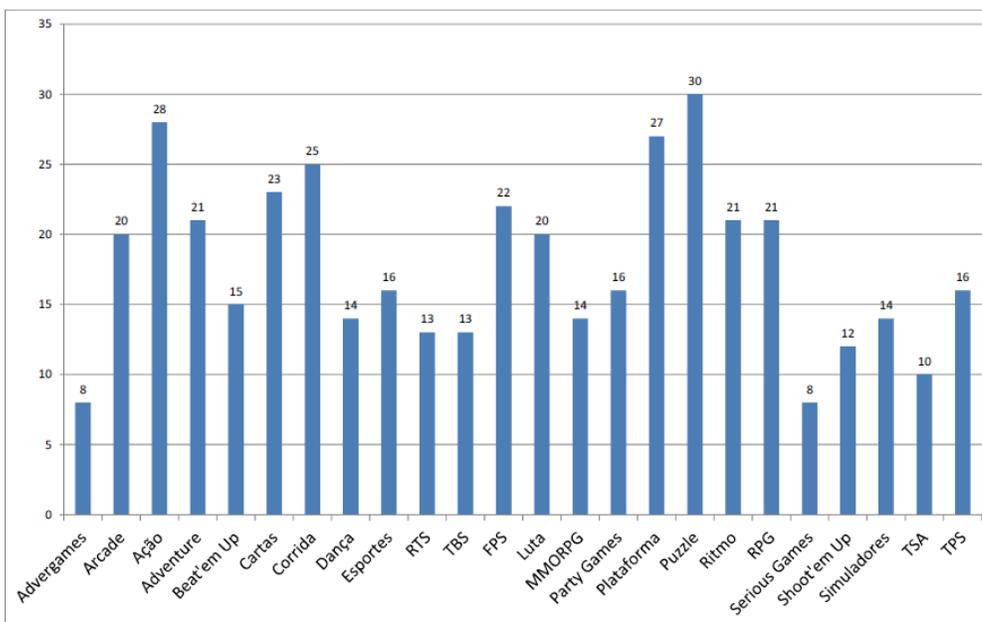


Figura 6.4: Frequência de gêneros conhecidos

Tabela 6.2: Resumo dos participantes - Idade, Horas e Gêneros conhecidos

	Média	Desvio padrão	Máximo	Mínimo
Idade	27,1429	6,1186	49	19
Idade - Casuais	27,6667	6,1453	42	19
Idade - Dedicados	25,1875	2,8802	31	20
Gêneros conhecidos	12,2000	7,1746	24	0
Horas/semana	6,6988	7,3476	30	0
Horas/semana - Casuais	2,0978	2,2240	10	0
Horas/semana - Dedicados	11,8750	7,9236	30	2

que algum acertasse. Isso fez com que a avaliação de modelo de jogador fosse um pouco prejudicada.

Outra característica problemática foi a taxa de inimigos abatidos. Quando os jogadores percebiam que não eram penalizados por evitar inimigos, apenas deixando de ganhar pontos, evitavam se arriscar a perder vidas procurando caminhos por onde pudessem fugir ou evitar, a fim de permanecerem vivos e chegar ao final do jogo. Essa atitude prejudicou a avaliação do modelo de jogador. Isso foi reforçado pela pontuação com pouco destaque no canto inferior direito da tela. Alguns jogadores relataram nem ter prestado atenção pois como os inimigos vinham de cima da tela, o único momento em que olhavam para a parte de baixo era quando precisavam escapar de alguns tiros ou evitar uma colisão.

Uma solução para ambos os problemas com características seria apresentar ao final de cada *wave* as estatísticas do jogador, como pontuação total e na *wave*, sua acurácia e o comparativo da variação com a avaliação anterior. Independente da aplicação na versão adaptativa ou não-adaptativa, esse comparativo pode funcionar como um fator motivacional para os jogadores, incentivando sua melhoria através de se arriscarem mais a abater os inimigos que simplesmente fugir dos tiros para chegar ao final. Tal artifício também proveria uma sensação de progresso ao fornecer um *feedback* de seu desempenho.

Há um porém nesta avaliação que fizemos: como os jogadores só jogaram uma vez cada versão, não foi possível avaliar com precisão se as mudanças causadas pela adaptatividade de dificuldade dinâmica poderiam ser de fato percebidas. Acreditamos que alguns jogadores depois de acostumados com o jogo começariam a notar a mudança nos parâmetros projetadas pelo sistema adaptativo, podendo procurar brechas no sistema para se favorecerem e quebrando a verossimilhança do mundo de jogo, o que prejudicaria o aproveitamento da experiência de jogo, como apontamos na seção 3.7.

Outras características poderiam ter sido incluídas na modelagem de jogador para tratar, por exemplo, da preferência de posicionamento do jogador

na tela. No caso, a modelagem do AIManager precisaria ser alterada para comportar características que não influenciassem na avaliação do modelo de jogador para nível de dificuldade como implementado. Isso seria útil para identificar que áreas eram preferidas em cada *wave* e o resultado poderia ser utilizado para balancear o posicionamento de inimigos dinamicamente ou para remodelagem do *level design*.

6.3 Resultados dos testes com jogadores

Em uma primeira análise, consultamos os valores do resultado dos testes pós-experiência de jogo aplicados ao final do teste com cada versão para compararmos as versões. Os resultados dos questionários pós-teste encontram-se nas Figuras B.2 para a versão 1 (adaptativa) e B.3 para a versão 2 (não adaptativa). O comparativo das somas e a média dos valores de cada questão correspondente aos fatores da escala de Calvillo-Gómez et al. (ver Tabela 5.2 para referência) encontram-se na Tabela 6.3.

Tabela 6.3: Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação ao total da população de testes

Comparativo das Escalas de CEGE							
		Versão 1		Versão 2			
		Fatores	Soma	Média	Soma	Média	Diferença %
Escala 1	Diversão	595	5,667	631	6,010	-5,71%	
	Frustração	155	2,214	162	2,314	-4,32%	
	ECEJ	6200	5,368	6156	5,330	0,71%	
	Titerismo	3726	5,069	3703	5,038	0,62%	
	Videogame	2474	5,890	2453	5,840	0,86%	
Escala 2	Controle	1842	5,848	1824	5,790	0,99%	
	Facilitadores	973	4,633	978	4,657	-0,51%	
	Posse	1106	4,514	1081	4,412	2,31%	
	Ambiente	1247	5,938	1231	5,862	1,30%	
	Game-play	1227	5,843	1222	5,819	0,41%	

Analisando os resultados da Tabela, pode-se notar que em ambas as versões foi verificada a presença do fator Diversão com baixa presença do fator Frustração, com certa vantagem para a versão não-adaptativa (diferença $\simeq 0.05$ em Diversão e $\simeq 0.04$ em Frustração). Verificando o resultado da diferença percentual na Tabela 6.3, pode-se notar que os outros fatores da escala 1 do questionário CEGE apresentam baixa diferença percentual. Para os fatores da escala 2, a versão adaptativa possui certa vantagem nos fatores Posse ($\simeq 0.02$) e Ambiente ($\simeq 0.01$) sobre a versão não-adaptativa.

Porém, se separarmos os jogadores dedicados dos casuais (ver Figuras B.4 e B.5), a diferença torna-se mais significativa, como mostra a Tabela 6.4. Pode-se perceber que a diferença no fator Frustração entre a versão 1 (adaptativa) e a versão 2 (não-adaptativa) é mais significativa ($\simeq -12\%$ menor).

Isso pode ser explicado pelas características do jogador dedicado, como mostradas na seção 2.2. Por possuir uma personalidade mais autotélica que a do jogador casual, em consonância com as características necessárias para atingir o estado de fluxo, como visto na seção 2.3, o jogador dedicado tende a buscar o desafio proporcionado pela variação de dificuldade proporcionada pela versão adaptativa e aceita melhor a mudança caso isso o leve a um estado de fluxo.

Tabela 6.4: Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação aos jogadores dedicados

Comparativo das Escalas de CEGE para jogadores dedicados							
		Versão 1		Versão 2			
		Fatores	Soma	Média	Soma	Média	Diferença %
Escala 1	Diversão	281	5,8542	285	5,9375	-1,40%	
	Frustração	74	2,3125	85	2,6563	-12,94%	
	ECEJ	2925	5,5398	2880	5,4545	1,56%	
	Titerismo	1775	5,2827	1756	5,2262	1,08%	
	Videogame	1150	5,9896	1124	5,8542	2,31%	
Escala 2	Controle	866	6,0139	859	5,9653	0,81%	
	Facilitadores	478	4,9792	477	4,9688	0,21%	
	Posse	529	4,7232	511	4,5625	3,52%	
	Ambiente	592	6,1667	566	5,8958	4,59%	
	Game-play	558	5,8125	558	5,8125	0,00%	

Já para os jogadores casuais (ver Tabela 6.5), os resultados mostram uma preferência pela versão não-adaptativa. Acreditamos que um dos principais motivos para os jogadores casuais terem esse resultado na avaliação da experiência de jogo seja devido à sua pouca familiaridade com o gênero de jogo apresentado em nossa implementação e à inexistência de níveis de dificuldade inferiores ao Fácil (*Easy*), que permitiriam uma melhor assimilação do estilo e das mecânicas do gênero *shoot'em up*. Além disso, Fortugno (Fortugno08) também afirma que as características que fazem um jogo ser interessante e prazeroso para jogadores dedicados são consideradas muito difíceis e desestimulantes por jogadores casuais (Fortugno08, p. 148), o que coloca a escolha de um jogo do gênero *shoot'em up* como um fator dificultante da experiência de jogo para jogadores casuais.

A Figura B.6 mostra o resultado dos testes, ilustrando qual foi a dificuldade inicial selecionada para cada versão e as dificuldades final, máxima e

Tabela 6.5: Comparativo do questionário de elementos centrais da experiência de jogo (CEGE) em relação aos jogadores casuais

Comparativo das escalas de CEGE para jogadores casuais							
		Versão 1		Versão 2			
		Fatores	Soma	Média	Soma	Média	Diferença %
Escala 1	Diversão	311	5,759259	336	6,222222	-7,44%	
	Frustração	73	2,027778	68	1,888889	7,35%	
	ECEJ	3145	5,294613	3157	5,314815	-0,38%	
	Titerismo	1869	4,944444	1870	4,94709	-0,05%	
	Videogame	1276	5,907407	1287	5,958333	-0,85%	
Escala 2	Controle	923	5,697531	910	5,617284	1,43%	
	Facilitadores	483	4,472222	489	4,527778	-1,23%	
	Posse	550	4,365079	546	4,333333	0,73%	
	Ambiente	637	5,898148	650	6,018519	-2,00%	
	Game-play	639	5,916667	637	5,898148	0,31%	

mínima percebidas para a versão adaptativa. Também é mostrado se o jogador terminou o jogo (i.e., venceu todas as *waves* de inimigos) e qual foi a máxima *wave* atingida. As *waves* são numeradas de 0 a 8, totalizando 9 *waves*.

Analisando a Figura B.6, percebe-se que a versão adaptativa teve um número maior de jogadores vitoriosos que a versão não-adaptativa (50% maior em relação à versão não-adaptativa). Dentre os vitoriosos na versão adaptativa, observa-se que houve variação de dificuldade, tanto para um nível mais difícil quanto para um nível mais fácil, em 7 dos 12 casos (58%).

Embora o resultado dos questionários mostre que o fator Frustração foi maior com a versão adaptativa e apesar de termos diluído o aprendizado ao realizar testes com versões alternadas entre os jogadores, muitas das afirmações feitas nas entrevistas colocam alterações que não eram esperadas. Em sua maioria, tais relatos referem-se a mudanças que não foram executadas, visto que as únicas mudanças entre as versões é a presença ou ausência do sistema adaptativo.

O jogador 34 citou que a versão 1 estava com mais atraso no tiro do jogador que a versão 2. Os jogadores 33 e 19 afirmaram que a música havia sido alterada. O jogador 32 achou que a versão 2 trazia mudanças na disposição dos inimigos enquanto jogava. O jogador 27 achou que houve uma diferença nos gráficos das versões, enquanto o jogador 15 afirmou que a versão 2 estava com menos cores. O jogador 18 achou a versão 2 mais difícil pelo aumento na dificuldade, embora ao se verificar a Figura B.6, percebe-se que não houve variação de dificuldade entre as versões jogadas.

Para esses casos, não temos uma resposta para o porquê da percepção de tais diferenças. Acreditamos que, por terem sido avisados de que jogariam

duas versões de um mesmo jogo, os jogadores procuraram diferenças mesmo onde não existiam.

Dentre os casos em que perceberam uma diferença na dificuldade, o jogador 1 afirmou que teve a impressão da versão 1 ter sido mais fácil, embora pela Figura B.6 percebe-se que a dificuldade foi elevada durante a partida. Atribuímos a esse caso o fator aprendizado, por ter sido essa versão a segunda que jogou.

7

Conclusão e trabalhos futuros

Pelos resultados apresentados na seção 6.3, nossa hipótese apresentada na seção 5.6.2 de que jogadores dedicados teriam maior facilidade em assimilar a experiência de jogo com a versão adaptativa foi confirmada. Esse fato é coerente com a teoria de fluxo apresentada na seção 2.3, uma vez que esperase que os jogadores dedicados tenham uma maior inclinação a alcançarem o estado de fluxo.

Os jogadores casuais apresentaram uma preferência maior pela versão não-adaptativa. Embora os motivos por essa preferência não tenham ficado muito evidentes, consideramos um dos fatores ser o gênero do jogo implementado, reconhecido por ser um gênero de nicho de jogadores dedicados, como enunciamos na seção 5.3. Como Schell afirma em relação aos jogadores e dificuldade:

“Mas é raro o jogador que é persistente o suficiente para vencer o jogo, dominando todos os níveis. A maioria dos jogadores acaba alcançando um nível em que eles passam tanto tempo na zona de frustração que desistem do jogo.” (Schell11, p. 121).

Consideramos os jogadores casuais importantes para estudos acadêmicos e de indústria por ser um público recentemente introduzido ao cenário dos jogos e que possui grande representatividade nos jogos por *download* (Fortugno08, p. 144). Acreditamos que técnicas de adaptatividade de dificuldade sejam interessantes para manter os jogadores casuais, além dos dedicados, em contato com o jogo por mais tempo ao reduzir a chance do jogador entrar na zona de frustração, porém nossa escolha de gênero de jogo para os testes não atendia às especificidades deste público, o que impactou na avaliação da experiência dos casuais com a versão adaptativa.

Poucos jogadores notaram a diferença na dificuldade ao jogar as versões adaptativa e não-adaptativa (jogadores P, 12 e 29, sendo assim dois jogadores dedicados e um casual, identificados na figura B.1) representando 8% da população de testes, como relatado na entrevista pós-teste. Essa baixa porcentagem é vista como um resultado positivo já que essa percepção não é desejada por levar à quebra da imersão, conforme visto na seção 5.4.

Realizar apenas um teste com cada versão prejudicou a avaliação de jogadores que tinham pouco ou nenhum contato com jogos do gênero implementado. Como visto na Figura 6.4, apenas doze de trinta e cinco participantes estavam familiarizados com o gênero *Shoot'em Up*. Embora o número de conhecedores do gênero *Arcade* seja de vinte participantes e o jogo implementado também pertença a este gênero, não é um tipo de jogo muito conhecido ou difundido. Uma sugestão para trabalhos futuros seria permitir que os testadores pudessem se familiarizar com o jogo por mais tempo e executar mais sessões de testes com cada jogador.

7.1

Principais contribuições

Como principais contribuições desse trabalho temos: a implementação e caso de uso do *framework* de adaptatividade dinâmica de Charles e Black (Charles04, Charles05); implementação eficiente de jogo adaptativo com aprendizagem *online*; a avaliação do uso de adaptatividade dinâmica com diferenciação de jogadores entre casuais e dedicados, mostrando que jogadores dedicados são mais receptivos a métodos adaptativos por definição; o conceito de jogos parcialmente adaptativos, como enunciamos na seção 3.1.

7.2

Trabalhos futuros

Para trabalhos futuros, pretendemos: ampliar as opções de jogo; refinar o *game-play* e o *level design* considerando as observações realizadas durante os testes e pelos comentários dos jogadores; revisitar a modelagem de jogador para incluir características que detectamos com os testes e uma maior granularidade de modelos considerando mais níveis de dificuldade intermediários; reavaliar e modificar a avaliação das características já implementadas, como por exemplo a preferência por determinada posição na tela por *wave* e o modo tiro contínuo em que alguns jogadores mantiveram o botão de tiro pressionado mesmo quando não havia inimigos em tela, prejudicando a avaliação da característica taxa de tiro.

Pretendemos também: implementar outras avaliações e técnicas de modelagem de jogador, como as usando redes neurais de (Yannakakis08a); investigar a aceitação e vantagens do uso de modelagem de jogador e adaptatividade dinâmica de dificuldade com outros gêneros de jogos, usando a biblioteca implementada AIManager, em especial com gêneros voltados ao público casual. Como analisamos na seção 6.3, o gênero *shoot'em up*, em especial o subgênero

bullet hell, não é muito conhecido por jogadores casuais dadas suas idiosincrasias que são voltadas a jogadores dedicados.

Em relação à metodologia, pretendemos explorar outras técnicas para avaliar a experiência de jogo, procurando identificar se o jogador atingiu o canal de fluxo. Como Schell afirma (Schell11, p. 122), algumas complicações são a necessidade de um tempo maior de observação do jogador, a diferença de resposta dos jogadores ao fluxo, no tocante à externalização de fatores fisiológicos e emocionais que possam identificar o fluxo e a própria possibilidade do jogo se tornar entediante ou frustrante com o tempo mesmo para o jogador que atinge o canal de fluxo.

Constitui ainda objeto de futuros trabalhos o uso de estatísticas mais elaboradas com os dados obtidos para mostrar qualidade, estabilidade e robustez dos resultados (t-student, p-value, standard deviation, ...). Correlações entre fatores também devem ser elaboradas, no futuro, para ajudar a entender ainda mais a experiência do jogador.

Uma melhoria imediata no algoritmo de jogo adaptativo seria considerar mais de uma característica de desempenho na regra de aprendizado (i.e., aplicar integralmente a regra de Widrow-Hoff). Esta melhoria também abriria novas direções de investigação buscando algoritmos *online* de aprendizagem mais elaborados. Neste sentido recomendamos averiguar o uso do modelo M5P (Wang97, Witten11) e testes com o *software* Weka¹.

Como meta de longo prazo, colocamos o interesse de buscar implementar adaptabilidade e entender a experiência de jogadores em jogos que automaticamente criem descrições semânticas de jogabilidade, de maneira que novos conteúdos (*quest*, *plots*, desafios, ...) possam ser automaticamente gerados. Esta é uma meta ambiciosa, também perseguida por pesquisadores de *story-telling* interativo.

¹<http://www.cs.waikato.ac.nz/ml/weka/>

Referências Bibliográficas

- [Andrade06] ANDRADE, G.; RAMALHO, G.; GOMES, A. S. ; CORRUBLE, V..
Dynamic game balancing: An evaluation of user satisfaction.
 In: AAAI CONFERENCE ON ARTIFICIAL INTELLIGENCE AND INTERAC-
 TIVE DIGITAL ENTERTAINMENT, p. 38, 2006. 3, 3, 3.6.1, 3.6.2
- [Apperley06] APPERLEY, T. H.. **Genre and game studies: toward a
 critical approach to video game genres.** Simulation & Gaming,
 37(1):6–23, Mar. 2006. 2.1.2
- [Atari72] ALCORN, A.. **Pong**, 1972. 2.1.2, 3.3.3
- [Atari77] DECUIR, J.; WAGNER, L.. **Combat**, 1977. 1.1
- [Bakkes09] BAKKES, S. C.; SPRONCK, P. H. ; JAAP VAN DEN HERIK, H.. **Op-
 ponent modelling for case-based adaptive game AI.** Entertainment
 Computing, 1(1):27–37, 2009. 4
- [Bartle96] BARTLE, R. A.. **Players who suit MUDs.** Journal of Online
 Environments, 1(1), 1996. 2.2.1, 4
- [Bernhaupt10] BERNHAUPT, R.. **User experience evaluation in enter-
 tainment.** In: Bernhaupt, R., editor, EVALUATING USER EXPERIENCE
 IN GAMES: CONCEPTS AND METHODS, p. 3–7. Springer, London, 1st
 edition, 2010. 5.3, 5.6.1
- [Bethesda06] STUDIOS, B. G.. **The Elder Scrolls IV: Oblivion**, Mar. 2006.
 3.4.2
- [Booth09] BOOTH, M.. **The AI systems of left 4 dead.**
[http://www.valvesoftware.com/publications/2009/ai_systems_of_l4d_mike
 _booth.pdf](http://www.valvesoftware.com/publications/2009/ai_systems_of_l4d_mike_booth.pdf), 2009. 1, 1.2, 3, 3.2.1
- [Brockmyer09] BROCKMYER, J. H.; FOX, C. M.; CURTISS, K. A.; MCBROOM,
 E.; BURKHART, K. M. ; PIDRUZNY, J. N.. **The development of
 the game engagement questionnaire: A measure of engagement
 in video game-playing.** Journal of Experimental Social Psychology,
 45(4):624–634, July 2009. 5.6, 5.6.2

- [Calvillo-Gómez09] CALVILLO GAMEZ, E. H.. **On the core elements of the experience of playing video games**. PhD thesis, UCL (University College London), 2009. 5.6.2
- [Calvillo-Gómez10] CALVILLO-GÁMEZ, E.; CAIRNS, P. ; COX, A.. **Assessing the core elements of gaming experience**. In: Bernhaupt, R., editor, **EVALUATING USER EXPERIENCE IN GAMES: CONCEPTS AND METHODS**, p. 47–71. Springer, London, 1st edition, 2010. 5.6, 5.6.1, 5.6.2, 5.7, 5.6.2, 5.6.2, 5.8, 5.2, A.2
- [Charles04] CHARLES, D.; BLACK, M.. **Dynamic player modeling: A framework for player-centered digital games**. In: PROC. OF THE INTERNATIONAL CONFERENCE ON COMPUTER GAMES: ARTIFICIAL INTELLIGENCE, DESIGN AND EDUCATION, p. 29–35, 2004. 1, 1.1, 1.4, 3.2.2, 3.3, 3.5, 3.7, 4, 5.2, 5.4.1, 5.6, 7.1
- [Charles05] CHARLES, D.; KERR, A.; MCNEILL, M.; MCALISTER, M.; BLACK, M.; KCKLICH, J.; MOORE, A. ; STRINGER, K.. **Player-centred game design: Player modelling and adaptive digital games**. In: PROCEEDINGS OF THE DIGITAL GAMES RESEARCH CONFERENCE, volumen 285, 2005. 1, 1.1, 3.2.2, 3.5, 4, 5.4.1, 7.1
- [Chen06] CHEN, J.; FULLERTON, T. J.. **Flow in games**. Master's thesis, University of Southern California, California, USA, 2006. 3.8, 3.4
- [Compile86a] COMPILE. **Zanac**, Nov. 1986. 3
- [Compile86b] COMPILE. **Zanac Instruction Manual**. Fujisankei Communications International, USA, 1986. 3
- [Cowley08] COWLEY, B.; CHARLES, D.; BLACK, M. ; HICKEY, R.. **Toward an understanding of flow in video games**. *Computers in Entertainment*, 6(2):1, July 2008. 1.1, 1.4, 2.2.1, 2.3, 2.2, 2.3, 2.3.1, 2.3, 3.7, 4
- [Crawford84] CRAWFORD, C.. **The art of computer game design**. Osborne/McGraw-Hill Berkley, 1984. 2.1.2, 2.1, 2.1.2, 2.1.2
- [Csikszentmihalyi90] CSIKSZENTMIHALYI, M.. **Flow : the psychology of optimal experience**. Harper & Row, New York, 1st edition, 1990. 1.1, 2.1.3, 2.3, 2.3.1
- [Demasi03a] DEMASI, P.; CRUZ, A. J. D. O.. **Estratégias Adaptativas e Evolutivas em Tempo Real para Jogos Eletrônicos**. Master's thesis, UFRJ, Rio de Janeiro, RJ, Brasil, 2003. 3.6.2

- [Demasi03b] DEMASI, P.; CRUZ, A. J. D. O.. **Evolução de agentes em tempo real para jogos eletrônicos de ação**. In: XI SIMPÓSIO BRASILEIRO DE SISTEMAS MULTIMÍDIA E WEB. WJogos - II WORKSHOP BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL, SALVADOR - BA, 4 E 5 DE NOVEMBRO DE 2003., Salvador, BA, Brazil, Nov. 2003. 3.6.2
- [Demasi03c] DEMASI, P.; CRUZ, A. J. D. O.. **Aprendizado de regras nebulosas em tempo real para jogos eletrônicos**. In: XI SIMPÓSIO BRASILEIRO DE SISTEMAS MULTIMÍDIA E WEB. WJogos - II WORKSHOP BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL, Salvador, BA, Brazil, Nov. 2003. 4.3
- [Dunniway08] DUNNIWAY, T.; NOVAK, J.. **Gameplay Mechanics**. Game Development Essentials. Delmar Cengage Learning, Clifton Park NY, 2nd edition, 2008. 3.4, 3.4.2
- [EAMobile11] MOBILE, E.. **FIFA 12**, Sept. 2011. 3.4.3
- [ESA12] ASSOCIATION, E. S.. **Essential facts about the computer and video game industry**. Technical report, ESA, USA, 2012. 5.6.2
- [Fortugno08] FORTUGNO, N.. **The strange case of the casual gamer**. In: Isbister, K.; Schaffer, N., editors, **GAME USABILITY: ADVICE FROM THE EXPERTS FOR ADVANCING THE PLAYER EXPERIENCE**, p. 143–158. Morgan Kaufman, Massachusetts, USA, 1st edition, 2008. 2.2.1, 10, 6.3, 7
- [Fullerton08] FULLERTON, T. J.; SWAIN, C. ; HOFFMAN, S.. **Game Design Workshop : A playcentric approach to creating innovative games**. Elsevier Morgan Kaufmann, Amsterdam, Boston, 2008. 2.2.1, 5.6.2
- [Gathering01] ENTERTAINMENT, R.. **Max payne**, July 2001. 1.2, 3.2.3
- [Gilleade04] GILLEADE, K. M.; DIX, A.. **Using frustration in the design of adaptive videogames**. In: PROCEEDINGS OF THE 2004 ACM SIGCHI INTERNATIONAL CONFERENCE ON ADVANCES IN COMPUTER ENTERTAINMENT TECHNOLOGY, p. 228–232, 2004. 1, 3.1, 3.7
- [Gularte10] GULARTE, D.. **Jogos eletrônicos: 50 anos de interação e diversão**. Ludo. 2AB Editora, 1st edition, 2010. 1.4, 2.1.2, 2.2
- [Gygax74] GYGAX, G.; ARNESON, D.. **Dungeons & Dragons**. Tactical Studies Rules Inc., 1974. 2.1.2, 9

- [Herik05] VAN DEN HERIK, H.; DONKERS, H. ; SPRONCK, P.. **Opponent modelling and commercial games**. Proceedings of IEEE, p. 15–25, 2005. 4
- [Hoonhout08] HOONHOUT, H.. **Let the game tester do the talking: Think aloud and interviewing to learn about the game experience**. In: Isbister, K.; Schaffer, N., editors, **GAME USABILITY: ADVICE FROM THE EXPERTS FOR ADVANCING THE PLAYER EXPERIENCE**, p. 65–77. Morgan Kauffman, 1st edition, 2008. 5.6.2, 5.6.2, 5.6.2
- [Houlette04] HOULETTE, R.. **Player modeling for adaptive games**. In: Rabin, S., editor, **AI GAME PROGRAMMING WISDOM 2**, p. 557–566. Charles River Media, Boston, MA, 1st edition, 2004. 3, 4, 4.1.2, 4.2, 4.3, 4.1, 5.2, 5.4.1, 5.5
- [Huizinga10] HUIZINGA, J.. **Homo Ludens: o jogo como elemento da cultura**. Estudos. Perspectiva, São Paulo, Brazil, 6 edition, 2010. 1.1, 2.1.1, 2.1, 2.2
- [Hunicke04a] HUNICKE, R.; LEBLANC, M. ; ZUBEK, R.. **MDA: a formal approach to game design and game research**. In: PROCEEDINGS OF THE AAAI-04 WORKSHOP ON CHALLENGES IN GAME AI, p. 1–5, 2004. 2.2.1, 3.3.2, 3.6.2
- [Hunicke04b] HUNICKE, R.; CHAPMAN, V.. **AI for dynamic difficulty adjustment in games**. In: PROCEEDINGS OF THE AAAI-04 WORKSHOP ON CHALLENGES IN GAME AI, p. 91–96, 2004. 3.3.2, 3.6.1
- [Hunicke05] HUNICKE, R.. **The case for dynamic difficulty adjustment in games**. In: PROCEEDINGS OF THE 2005 ACM SIGCHI INTERNATIONAL CONFERENCE ON ADVANCES IN COMPUTER ENTERTAINMENT TECHNOLOGY, volumen 256, p. 429 – 433. ACM, 2005. 1, 1.1, 3, 3.2.2, 3.3.2, 3.6.1
- [Ibáñez11] IBÁÑEZ, J.; DELGADO-MATA, C.. **Adaptive two-player video-games**. Expert Systems with Applications, 38:9157–9163, Aug. 2011. 3.3.3, 3.6.1, 5.6
- [Ierusalimschy06] IERUSALIMSKY, R.. **Programming in Lua**. Lua.org, Rio de Janeiro, 2nd edition, 2006. 5.3
- [IntiCreates02] CREATES, I.. **Mega Man Zero**, Apr. 2002. 3.4.1
- [IntiCreates03] CREATES, I.. **Mega Man Zero 2**, May 2003. 3.4.1

- [IntiCreates04] CREATES, I.. **Mega Man Zero 3**, Apr. 2004. 3.4.1
- [IntiCreates05] CREATES, I.. **Mega Man Zero 4**, Apr. 2005. 3.4.1
- [Johnson04] JOHNSON, S.. **Adaptive AI: a practical example**. In: Rabin, S., editor, **AI GAME PROGRAMMING WISDOM 2**, p. 639–647. Charles River Media, Hingham, Massachusetts, 1st edition, 2004. 3.6.2
- [Juul03] JUUL, J.. **The game, the player, the world: Looking for a heart of gameness**. **Level Up: Digital Games Research Conference Proceedings**, p. 30–45, 2003. 2.1.1
- [Koeffel10] KOEFFEL, C.; HOCHLEITNER, W.; LEITNER, J.; HALLER, M.; GEVEN, A. ; TSCHELIGI, M.. **Using heuristics to evaluate the overall user experience in games and advanced interaction**. In: Bernhaupt, R., editor, **EVALUATING USER EXPERIENCE IN GAMES: CONCEPTS AND METHODS**, p. 233–256. Springer, London, 1st edition, 2010. 5.6
- [Koster04] KOSTER, R.. **A Theory of Fun for Game Design**. Paraglyph Press, Scottsdale AZ, 1st edition, 2004. 1.1, 2.1.1, 2.2, 2.2.1, 2.2.1, 2.3.1, 3.1, 3.2.3
- [Kultima10] KULTIMA, A.; STENROS, J.. **Designing games for everyone**. In: **PROCEEDINGS OF THE ACM FuturePlay 2010**, p. 66–73, Canada, 2010. ACM Press. 2.2.1
- [Lankes10] LANKES, M.; BERNHAUPT, R. ; TSCHELIGI, M.. **Evaluating user experience factors using experiments: Expressive artificial faces embedded in contexts**. In: Bernhaupt, R., editor, **EVALUATING USER EXPERIENCE IN GAMES: CONCEPTS AND METHODS**, p. 165–183. Springer, London, 1st edition, 2010. 5.6
- [Lopes11] LOPES, R.; BIDARRA, R.. **Adaptivity challenges in games and simulations: A survey**. **IEEE Transactions on Computational Intelligence and AI in Games**, 3(2):85–99, June 2011. 1, 3, 3, 2, 3.2.3
- [Machado11a] MACHADO, M.; FANTINI, E. ; CHAIMOWICZ, L.. **Player modeling: Towards a common taxonomy**. In: **COMPUTER GAMES (CGAMES), 2011 16TH INTERNATIONAL CONFERENCE ON**, p. 50–57, 2011. 1.4, 4.1, 4.1, 4.1.1, 4.2, 4.3
- [Machado11b] MACHADO, A. F. D. V.; DUARTE, C. G.; ROGÉRIO, H. E.; PAULO, L. M. D.; PAULA, W. C. S. D.; CLUA, E. W. G. ; ROGERS, T. J.. **Dynamic difficulty balancing of strategy games through player**

- adaptation using top culling.** In: PROCEEDINGS OF SBGames 2011, Salvador, Bahia, Brazil, Nov. 2011. 3.6.3, 5.6
- [Machado11c] MACHADO, M.; FANTINI, E. ; CHAIMOWICZ, L.. **Player modeling: What is it? how to do it?** In: PROCEEDINGS OF SBGames 2011, Salvador, Bahia, Brazil, Nov. 2011. 4.2
- [Malone84] MALONE, T. W.. **Heuristics for designing enjoyable user interfaces: Lessons from computer games.** In: HUMAN FACTORS IN COMPUTER SYSTEMS, p. 1–12, 1984. 3.6.3
- [Mallory12] MALLORY, S.. **Social gaming and the bartle archetypes.** http://gamasutra.com/blogs/SteveMallory/20120413/168507/Social_Gaming_and_the_Bartle_Archetypes.php, Apr. 2012. 2.2.1
- [Missura09] MISSURA, O.; GARTNER, T.. **Player modeling for intelligent difficulty adjustment.** In: DISCOVERY SCIENCE, p. 197–211, 2009. 4
- [Missura11] MISSURA, O.; GÄRTNER, T.. **Predicting dynamic difficulty.** In: PROCEEDINGS OF THE NINTH WORKSHOP ON MINING AND LEARNING WITH GRAPHS, San Diego, CA, 2011. 3, 3.6.3
- [Miyamoto81] MIYAMOTO, S.; YOKOI, G.. **Donkey Kong**, July 1981. 2.1.2
- [Moffett10] MOFFETT, J. P.; RICH, C. ; BECK, J.. **Applying Causal Models to Dynamic Difficulty Adjustment in Video Games.** Master's thesis, Worcester Polytechnic Institute, Massachusetts, USA, May 2010. 3.6.2
- [Montfort09] MONTFORT, N.; BOGOST, I.. **Racing the Beam: the Atari video computer system.** Número 1 em Platform Studies. MIT Press, Cambridge, Massachusetts, 1 edition, 2009. 1.1, 14
- [Morrison81] MORRISON, H. J.. **Adaptive microcomputer controlled game**, Aug. 1981. 3
- [Nacke10] NACKE, L. E.; LINDLEY, C. A.. **Affective ludology, flow and immersion in a first- person shooter: Measurement of player experience.** Computing Research Repository, abs/1004.0248, 2010. 5.6
- [Nintendo85] DEPARTMENT, N. C.. **Super Mario Bros.**, Sept. 1985. 8
- [Nintendo96] NINTENDO EAD. **Mario Kart 64**, Dec. 1996. 3.2.2
- [Noon09] NOON, A.; PHILIPPIDES, A. ; THOMPSON, A.. **Dynamic difficulty adjustment in games: A neuroevolutionary approach.** Master's thesis, University of Sussex, East Sussex, England, Sept. 2009. 3.6.2

- [Novak11] NOVAK, J.. **Desenvolvimento de Games**. Cengage Learning, Brazil, 2011. 2.2.1, 2.2.1, 2.2.1, 3, 5.3
- [Parnell09] PARNELL, M. J.; BERTHOUSE, N. ; BRUMBY, D.. **Playing with Scales: Creating a Measurement Scale to Assess the Experience of Video Games**. Master's thesis, University College London, England, 2009. 5.6.2
- [Pedersen09] PEDERSEN, C.; TOGELIUS, J. ; YANNAKAKIS, G.. **Modeling Player Experience in Super Mario Bros**. In: COMPUTATIONAL INTELLIGENCE AND GAMES, 2009. CIG 2009. IEEE SYMPOSIUM ON, p. 132–139, 2009. 4.3
- [Persson08] PERSSON, M.. **Infinite Mario Bros**. <http://www.mojang.com/notch/mario/>, 2008. 3.3.1
- [Poole07] POOLE, S.. **Trigger Happy: Videogames and the entertainment revolution**. Publicado online pelo autor, 2007. 2.1.1
- [Ranhel09] RANHEL, J. A.. **O conceito de jogo e os jogos computacionais**. In: Santaella, L.; Feitoza, M., editors, MAPA DO JOGO: A DIVERSIDADE CULTURAL DOS GAMES, Profissional, p. 3–22. Cengage Learning, São Paulo, Brazil, 1st edition, 2009. 2.1.1, 2.1
- [RockinAndroid10] ANDROID, R.. **GundeadliGne**, June 2010. 3.2.4
- [Rollings03] ROLLINGS, A.; ADAMS, E.. **Andrew Rollings and Ernest Adams on game design**. New Riders, Berkeley Calif., 2003. 2.1.2
- [Salen03] SALEN, K.; ZIMMERMAN, E.. **Rules of play : game design fundamentals**. MIT Press, Cambridge Mass., 2003. 2.1.1, 2.1
- [Salen06] Salen, K.; Zimmerman, E., editors. **The game design reader : a rules of play anthology**. MIT Press, Cambridge Mass., 2006. 2.1.1
- [Schwab04] SCHWAB, B.. **AI Game Engine Programming**. Charles River Media,, Hingham, MA, 2004. 3.7
- [Schell11] SCHELL, J.. **A Arte de Game Design: o livro original**. Media Technology. Elsevier, 1st edition, 2011. 2.2, 2.2.1, 2.2.1, 2.3, 2.3, 12, 2.3.1, 15, 3, 3.7, 5.6, 7, 7.2
- [Smith11] SMITH, A.; LEWIS, C.; HULLETT, K.; SMITH, G. ; SULLIVAN, A.. **An inclusive taxonomy of player modeling**. Technical Report UCSC-SOE-11-13, Center for Games and Playable Media, University of California, Santa Cruz, Santa Cruz, CA, 2011. 4.1, 4.2

- [Sony05] SCE STUDIOS SANTA MONICA. **God of War**, Mar. 2005. 3.4.3
- [Spronck06] SPRONCK, P.; PONSEN, M.; SPRINKHUIZEN-KUYPER, I. ; POSTMA, E.. **Adaptive game AI with dynamic scripting**. Machine Learning, 63(3):217–248, Mar. 2006. 3.6.2, 3.6.3
- [Stewart11] STEWART, B.. **Personality and play styles: A unified model**. http://www.gamasutra.com/view/feature/6474/personality_and_play_styles_a_.php, Sept. 2011. 2.2.1, 4
- [Thomas07] THOMAS, L. M.. **Zanac review - wii review at IGN**. <http://wii.ign.com/articles/839/839646p1.html>, Dec. 2007. 3
- [Valve98] CORPORATION, V.. **Half-Life**, Nov. 1998. 3.3.2
- [Valve04] CORPORATION, V.. **Half-Life 2**, Nov. 2004. 3.3.2
- [Valve08] VALVE CORPORATION. **Left 4 Dead**, Nov. 2008. 1.2, 3.2.1, 4.1.1
- [Valve09] VALVE CORPORATION. **Left 4 Dead 2**, Nov. 2009. 1.2, 3.2.1
- [Wang97] WANG, Y.; WITTEN, I. H.. **Induction of model trees for predicting continuous classes**. In: POSTER PAPERS OF THE 9TH EUROPEAN CONFERENCE ON MACHINE LEARNING. Springer, 1997. 7.2
- [Weber10a] WEBER, B.. **Infinite Adaptive Mario**. http://users.soe.ucsc.edu/~bweber/dokuwiki/doku.php?id=infinite_adaptive_mario, 2010. 3.3.1
- [Weber10b] WEBER, B.. **Results 2010 - Mario AI Championship 2010**. <http://www.marioai.org/results-2010>, Aug. 2010. 3.3.1
- [Weber10c] WEBER, B.. **Proceedings of the IEEE conference on computational intelligence and games (CIG) 2010 - Competitions**. <http://game.itu.dk/cig2010/proceedings/wp-content/competitions.html>, Aug. 2010. 3.3.1
- [Widrow88] WIDROW, B.; HOFF, M. E.. **Adaptive switching circuits**. In: Anderson, J. A.; Rosenfeld, E., editors, **NEUROCOMPUTING: FOUNDATIONS OF RESEARCH**, p. 123–134. MIT Press, Cambridge, MA, USA, 1988. 5.5
- [Witten11] WITTEN, I. H.; FRANK, E. ; HALL, M. A.. **Data mining : practical machine learning tools and techniques**. Morgan Kaufmann, Burlington, MA, 2011. 7.2

- [Xavier10] XAVIER, G.. **A condição eletrolúdica: Cultura visual nos jogos eletrônicos**. 2AB Editora, Teresópolis, Brasil, 2010. 2.1.1
- [Yannakakis05] YANNAKAKIS, G.; MARAGOUidakis, M.. **Player modeling impact on player's entertainment in computer games**. In: Ardissono, L.; Brna, P. ; Mitrovic, A., editors, **USER MODELING 2005: 10TH INTERNATIONAL CONFERENCE**, volumen 3538 de **Lecture notes in computer science**, p. 74–78, Edingburgh, Scotland, UK, July 2005. Springer. 4, 4.3
- [Yannakakis08a] YANNAKAKIS, G. N.; HALLAM, J.. **Real-time adaptation of augmented-reality games for optimizing player satisfaction**. In: **COMPUTATIONAL INTELLIGENCE AND GAMES, 2008. CIG'08. IEEE SYMPOSIUM ON**, p. 103–110, 2008. 3.6.3, 4.3, 7.2
- [Yannakakis08b] YANNAKAKIS, G. N.. **How to model and augment player satisfaction: A review**. In: **PROCEEDINGS OF THE 1ST WORKSHOP ON CHILD, COMPUTER AND INTERACTION, CHANIA, CRETE, ACM PRESS, 2008**. 4.3
- [thatgamecompany06] THATGAMECOMPANY. **fLOW**, Apr. 2006. 7, 3.8

A

Questionários

A.1

Pré-teste

Os seguintes dados foram levantados sobre cada participante no questionário pré-teste:

1. Idade
2. Sexo
3. Se o entrevistado já havia jogado jogos eletrônicos antes do teste
4. Se o entrevistado se considera um jogador casual ou dedicado
5. Quantas horas por semana o entrevistado dedica a jogos
6. Os gêneros de jogos com os quais o entrevistado estava familiarizado, dentre as opções:
 - (a) Advergame
 - (b) Arcade
 - (c) Ação
 - (d) Adventure
 - (e) Beat'em up
 - (f) Cartas
 - (g) Corrida
 - (h) Dança
 - (i) Esportes
 - (j) Estratégia em Tempo Real
 - (k) Estratégia em Turnos
 - (l) First Person Shooter
 - (m) Luta

- (n) MMORPG
- (o) Party Games
- (p) Plataforma
- (q) Puzzle
- (r) Ritmo
- (s) RPG
- (t) Serious Games
- (u) Shoot'em Up
- (v) Simuladores
- (w) Tactical Stealth Action
- (x) Third Person Shooter

A.2

Pós-experiência de jogo

As seguintes questões foram traduzidas por nós do modelo de Cálvillo-Gámez (Cálvillo-Gámez¹⁰, p. 69-70) e foram avaliadas no questionário pós-experiência de jogo em uma escala Likert de sete pontos, variando de Discordo Muito (1) a Concordo Muito (7). O asterisco (*) indica itens formulados negativamente.

A correlação de cada questão aos fatores de experiência de jogo avaliados pelo questionário podem ser verificados na Tabela 5.2.

1. Eu me diverti com o jogo.
2. Eu me senti frustrado ao final do jogo.
3. Eu me senti frustrado enquanto jogava.
4. Eu gostei do jogo.
5. Eu jogaria novamente esse jogo.
6. Eu estava dominando o jogo.
7. Os controles responderam como eu esperava.
8. Eu me lembro das ações que os controles faziam.
9. Eu pude ver na tela tudo que eu precisava durante o jogo.
10. *A visão que eu tinha do jogo estragou a minha experiência com o jogo.

11. Eu sabia o que tinha que fazer para vencer no jogo.
12. *Houve momentos em que eu nada fazia no jogo.
13. Eu gostei da aparência do jogo.
14. Os gráficos do jogo eram fáceis de entender.
15. *Eu não gosto desse tipo de jogo.
16. Eu gosto de passar muito tempo jogando este jogo.
17. Eu me senti entendiado jogando dessa vez.
18. *Normalmente não escolho esse tipo de jogo.
19. *Eu não tinha uma estratégia para vencer no jogo.
20. O jogo constantemente me motivava a continuar jogando.
21. Eu senti que o que acontecia no jogo era por minha causa.
22. Eu desafiei a mim mesmo mesmo que o jogo não exigisse isso.
23. Eu joguei com minhas próprias regras.
24. *Eu me senti culpado pelas ações do jogo.
25. Eu sabia como manipular o jogo para avançar.
26. Os gráficos estavam apropriados para o tipo de jogo.
27. Os efeitos sonoros do jogo eram apropriados.
28. *Eu não gostei da música do jogo.
29. Os gráficos do jogo estavam relacionados ao conceito.
30. Os gráficos e efeitos sonoros estavam relacionados.
31. O som do jogo afetou a forma como eu estava jogando.
32. *O jogo foi injusto.
33. Eu entendi as regras do jogo.
34. O jogo foi desafiador.
35. O jogo foi difícil.
36. O conceito do jogo era interessante.

37. *Eu não gostei do conceito do jogo.

38. Eu sabia todas as ações que podia fazer no jogo.

A.3 **Entrevista**

Os tópicos selecionados para abordagem na entrevista pós-testes estão elencados a seguir, juntamente com algumas perguntas elaboradas previamente ao teste para direcionar a aquisição de dados.

- Diferenças percebidas entre os jogos.
- Qual versão achou mais desafiadora.
- O que sentiu falta.
- O que poderia ser melhorado.
- Padrão de ação detectado no jogo. Por exemplo, manter-se à esquerda, evitar atirar nos inimigos. . .
- Opinião sobre sistema adaptativo de dificuldade.

B
Tabelas dos testes

ID	Ordem	Versão	Questões																																															
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38										
P	A	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38											
1	B	A	1	7	7	5	7	7	7	3	5	7	7	6	7	7	5	1	1	1	6	7	5	1	7	7	7	7	6	7	7	7	7	7	2	7	7	5	3	7	7									
2	A	1	6	2	1	6	5	4	6	7	5	6	7	7	3	4	2	1	1	2	5	1	6	2	6	5	6	7	7	6	6	6	6	6	6	7	7	6	4	5	6	3								
3	B	A	1	6	7	1	1	7	7	7	7	6	7	7	7	3	1	6	7	7	7	7	7	6	5	1	6	5	7	7	7	7	6	7	7	6	7	6	4	6	6	7								
4	A	1	3	3	2	3	6	7	7	7	7	7	4	6	7	7	5	3	5	4	7	2	1	7	6	4	1	4	7	7	7	7	6	6	5	7	7	6	7	2	3	7	7							
5	B	A	1	7	7	1	1	7	7	5	7	7	2	7	7	6	1	1	5	1	4	3	2	2	1	1	6	7	7	6	4	6	6	5	7	7	6	3	6	7	7	7	7							
6	A	1	5	4	1	5	2	1	7	7	4	2	7	6	7	1	1	4	1	1	5	3	7	1	2	1	1	7	7	7	3	7	7	4	7	7	4	7	7	3	3	7	7							
7	B	A	1	6	7	1	1	6	7	6	4	7	6	5	7	7	4	1	6	1	5	4	4	3	4	6	4	8	6	6	5	5	2	6	6	4	7	7	2	2	6	6	3							
8	A	1	6	1	1	6	5	6	4	7	5	7	6	5	7	7	4	3	2	2	4	4	3	4	8	4	8	6	6	5	5	2	6	6	5	2	6	6	4	2	2	6	6	3						
9	B	A	1	5	4	2	5	4	6	6	6	6	7	6	3	6	7	5	2	6	4	6	5	2	7	7	7	7	5	5	6	6	6	6	5	7	7	7	7	7	2	2	6	6	3					
10	A	1	7	1	1	1	1	7	7	7	7	7	1	4	4	1	1	5	1	1	3	3	6	1	7	6	4	4	1	4	4	1	4	4	1	7	7	7	7	7	7	4	1	4	6	7				
11	B	A	1	4	5	9	2	5	7	6	7	7	6	7	7	7	7	4	1	2	6	5	7	9	2	7	6	4	7	7	7	7	6	4	7	7	1	7	7	6	9	6	7	7	7					
12	A	1	7	5	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7					
13	B	A	1	7	4	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7					
14	A	1	5	5	5	7	7	7	7	7	6	4	7	7	5	6	2	5	1	4	6	6	7	4	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6				
15	B	A	1	6	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7				
16	A	1	6	3	3	6	3	6	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7				
17	B	A	1	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7			
18	A	1	7	5	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7			
19	B	A	1	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7			
20	A	1	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
21	B	A	1	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
22	A	1	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
23	B	A	1	7	2	2	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
24	A	1	6	1	1	7	7	1	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
25	B	A	1	7	1	1	6	6	3	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7		
26	A	1	4	2	2	4	5	5	6	7	6	4	5	2	5	6	6	4	4	6	6	3	3	5	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
27	B	A	1	7	3	1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
28	A	1	5	3	1	5	5	4	6	6	6	6	4	4	7	3	2	2	3	2	4	4	5	1	5	7	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	
29	B	A	1	6	1	1	6	7	3	2	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
30	A	1	5	6	2	4	2	4	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
31	B	A	1	7	2	2	7	7	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
32	A	1	9	3	1	7	7	4	6	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
33	B	A	1	4	6	2	2	2	1	7	4	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
34	A	1	4	5	6	2	2	2	1	7	4	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
Soma:			189	32	49	203	193	138	205	233	218	251	200	304	187	233	191	244	61	147	135	177	155	148	151	143	195	213	224	169	224	227	160	228	237	305	159	188	313	300										
Média:			6,86	2,629	1,600	5,800	5,14	3,943	5,657	6,657	6,229	6,314	6,268	5,829	6,629	6,657	5,457	4,114	1,743	4,200	3,657	5,286	4,171	3,571	4,086	5,571	6,086	6,400	5,686	6,400	6,466	4,571	6,457	6,771	5,657	4,543	5,371	6,057	5,643											

Figura B.2: Resultado dos questionários de experiência de jogo para versão adaptativa

ID	Dificuldade					Terminou		Wave atingida	
	Inicial		Final	Máxima	Mínima	Versão 1	Versão 2	Versão 1	Versão 2
	Versão 1	Versão 2	Versão 1						
P	Easy	Easy	Medium	Medium	Easy	x	x	8	8
1	Easy	Easy	Easy	Medium	Easy	x	x	8	8
2	Medium	Medium	Medium	Medium	Medium			3	4
3	Medium	Easy	Easy	Medium	Easy			4	4
4	Easy	Easy	Easy	Easy	Easy			6	4
5	Medium	Medium	Easy	Medium	Easy	x	x	8	8
6	Easy	Easy	Easy	Easy	Easy			3	4
7	Easy	Easy	Easy	Easy	Easy			5	3
8	Medium	Hard	Medium	Medium	Easy	x		8	3
9	Medium	Medium	Medium	Medium	Medium			4	1
10	Easy	Easy	Medium	Medium	Easy			4	4
11	Easy	Easy	Easy	Easy	Easy			1	1
12	Easy	Easy	Easy	Medium	Easy	x		8	4
13	Medium	Medium	Easy	Medium	Easy	x		8	6
14	Easy	Easy	Easy	Easy	Easy			1	4
15	Medium	Medium	Medium	Medium	Medium	x		8	6
16	Medium	Medium	Medium	Medium	Easy			3	6
17	Easy	Easy	Medium	Medium	Easy			5	7
18	Medium	Medium	Medium	Medium	Medium	x	x	8	8
19	Hard	Hard	Medium	Hard	Medium			6	4
20	Easy	Easy	Easy	Easy	Easy			6	5
21	Easy	Easy	Easy	Easy	Easy			6	5
22	Easy	Easy	Easy	Easy	Easy			5	4
23	Easy	Easy	Easy	Easy	Easy			5	3
24	Easy	Easy	Easy	Easy	Easy			3	5
25	Easy	Easy	Easy	Medium	Easy			4	4
26	Medium	Medium	Easy	Medium	Easy	x	x	8	8
27	Medium	Medium	Medium	Medium	Medium	x	x	8	8
28	Medium	Medium	Medium	Medium	Medium	x	x	8	8
29	Hard	Hard	Medium	Hard	Medium			8	5
30	Easy	Easy	Medium	Medium	Easy	x	x	8	8
31	Easy	Easy	Easy	Easy	Easy			7	4
32	Easy	Easy	Easy	Easy	Easy			3	3
33	Easy	Easy	Easy	Easy	Easy			4	4
34	Medium	Medium	Medium	Medium	Medium			3	7
Total:						12	8		

Figura B.6: Jogadores e desempenho nas versões