Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Comparação de ajustes dinâmicos de dificuldade aplicados a diferentes elementos de jogo

Eduardo de Azevedo dos Santos
Iago Lobo Ribeiro de Moraes

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientadora
Prof.a Dr.a Carla Denise Castanho

Brasília
2021

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Comparação de ajustes dinâmicos de dificuldade aplicados a diferentes elementos de jogo

Eduardo de Azevedo dos Santos
Iago Lobo Ribeiro de Moraes

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof.a Dr.a Carla Denise Castanho (Orientadora)
CIC/UnB

Prof. Dr. Tiago Barros Pontes e Silva
DIN/UnB

Prof. Dr. Mauricio Miranda Sarmet
IFPB

Prof. Dr. Marcelo Grandi Mandelli
Coordenador do Bacharelado em Ciência da Computação

Brasília, 21 de maio de 2021

Dedicatória

Eu, Eduardo, dedico este trabalho à minha família, a qual sempre esteve presente incondicionalmente ao meu lado. À minha namorada, companheira a qual escolhi para a vida. Dedico também aos amigos e professores que conheci e que me ajudaram ao longo de minha trajetória universitária.

Eu, Iago, dedico este trabalho à minha família, que me apoiou com seu imenso amor e forneceu as ferramentas necessárias para chegar onde cheguei. Dedico também aos amigos e professores que fizeram a diferença na minha jornada pessoal, profissional e acadêmica durante meus estudos universitários.

Agradecimentos

Agradecemos a nossa orientadora e professora Carla Denise Castanho, aos professores Mauricio Sarmet e Tiago Barros por participarem da banca avaliadora, ao mestrando do Programa de Pós-Graduação em Informática da Universidade de Brasília Marcos Paulo Cayres Rosa pela assistência, colaboração e discussões, assim como a todos os voluntários que participaram dos testes realizados.

Resumo

Conforme a indústria de videogames cresce, há o surgimento de novos cenários, sendo essencial manter os jogos divertidos para distintos perfis de consumidor, os quais podem possuir diversos níveis de habilidades e preferências. Nesse contexto é relevante a implementação de um sistema de dificuldade adaptável ao jogador, assim oferecendo um desafio condizente ao jogador, evitando que o jogo se torne tedioso por estar muito fácil ou muito difícil. Diversos sistemas já foram modelados para lidar com essa questão, dentre eles alguns específicos ao gênero de jogo plataforma. No âmbito deste trabalho, objetiva-se comparar a eficiência de um sistema de ajuste dinâmico de dificuldade (ADD) por desempenho aplicado a elementos diferentes de um jogo de plataforma. Foram comparados um total de quatro casos: sem aplicação de ADD, aplicação de ADD às plataformas, aplicação de ADD ao pulo e aplicação de ADD a ambos. Para tanto, utilizou-se uma evolução do trabalho de Rosa [1], sobre o qual foram implementadas as devidas modificações e para a condução de testes remotos cujo os dados foram coletados através de um Google Forms. A análise desses dados revelou conclusões multifacetadas, com o caso que melhor permite o estado de fluxo sendo o caso de ADD combinado, seguido pelo ADD de plataforma.

Palavras-chave: jogos eletrônicos, ajuste dinâmico de dificuldade, plataforma 2D

Abstract

As the video game industry grows, new scenarios are emerging, so it is essential to keep games fun for different consumer profiles, who may have varying levels of skills and preferences. In this context, it is relevant to implement a difficulty system that adapts to the player, thus offering a suitable challenge to the player, preventing the game from becoming tedious because it is too easy or too difficult. Several systems have already been modeled to deal with this issue, among them some specific to the platform game genre. In the scope of this work, the objective is to compare a system of dynamic difficulty adjustment (DDA) by performance applied to different elements of a platform game, to find out which one has the best efficiency in keeping the player in the flow state. A total of four cases were compared: without application of ADD, application of ADD to platforms, application of ADD to the jump and application of ADD to both. To this end, an evolution of the work of Rosa [1] was used, on which the necessary modifications were implemented and for conducting remote tests whose data were collected through a Google Forms. The analysis of these data revealed multifaceted conclusions, with the case that best allows the flow state being the case of combined ADD, followed by the platform ADD.

Keywords: video game, dynamic difficult adjustment, 2D platform

Sumário

1 Introdução **1**
1.1 Definição do problema 1
1.1.1 Objetivos 2
1.1.2 Estruturação 2

2 Fundamentação teórica **4**
2.1 Jogo 4
2.1.1 Taxonomia dos jogos 7
2.1.2 Categorização de jogos de plataforma 7
2.2 Jogador 9
2.2.1 Classificação de jogador 9
2.3 Fluxo 10
2.3.1 Fluxo em jogos 12
2.4 A dificuldade em jogos 14
2.4.1 Adaptatividade 16
2.4.2 Ajuste dinâmico de dificuldade 17
2.4.3 Dificuldade em jogos de plataforma 19
2.5 A experiência do jogador 22

3 Proposta de ADD em característica da personagem **23**
3.1 Trabalho de Rosa 23
3.1.1 O Jogo 24
3.1.2 Coleta de dados do trabalho de Rosa 31
3.1.3 Medição da dificuldade do ADD do trabalho de Rosa 34
3.1.4 Sensor do ADD afetivo do trabalho de Rosa 41
3.1.5 ADD híbrido do trabalho de Rosa 42
3.1.6 Ajuste dos componentes pela dificuldade do trabalho de Rosa 45
3.2 Proposta 47

4 Testes e resultados **51**
4.1 Testes 51
4.1.1 Descrição dos níveis e execução dos testes 52
4.2 Resultados 53

5 Considerações Finais **62**
5.1 Conclusões 62
5.2 Trabalhos futuros 64

Referências **66**

Apêndice **69**

A Questionário **70**
A.1 Dados sociodemográficos 70
A.2 Jogo 71

Lista de Figuras

2.1 Imagens dos jogos *Super Mario Bros.* (A), *Little Big Planet* (B) e *Sonic the Hedgehog* (C) [2]. 9
2.2 Diagrama do canal de fluxo, adaptado por Rosa [1], contendo as três dimensões de uma experiência: ansiedade, fluxo e tédio [3]. 13
2.3 As oito dimensões da experiência [4]. Consiste no diagrama do modelo de flutuação da experiência tal que o ponto médio do gráfico corresponde a média de habilidades e desafios de uma pessoa para uma dada atividade [3]. 13
2.4 a) Experiência do estado de fluxo em um jogador; b) Jogador ao encontrar entropias psíquicas; c) Diferentes jogadores e zonas de fluxo (retirado de [1]). 14

3.1 Captura de tela da versão original do jogo *The Explorer: 2D*. 25
3.2 HUD contendo vidas, itens coletáveis e interface do sistema de diálogo da versão original do jogo *The Explorer: 2D*. 27
3.3 Arquivo de entrada para a geração da fase e o resultado visual, adaptada de [1]. 28
3.4 Principais componentes categorizados do jogo, conforme ilustrado ao usuário durante a explicação dos comandos no início do jogo [1]. 30
3.5 Tipos de plataformas. As setas apontam para a plataforma destino do pulo, com seu nome correspondente associado. O caso de exceção Esquerda Direita tem setas apontando para ambas as direções e se refere a um buraco ou abismo no lugar de uma plataforma [1]. 33
3.6 Demonstração de desafios em jogos do gênero de plataforma [1]. 36
3.7 Conceito de margem de erro demonstrado visualmente [5]. 39
3.8 (a) Categorização do pulo conforme sua posição de destino (P_1) referente à origem (P_0); (b) Valores de correção para ajustar o pulo de uma trajetória [1]. 41
3.9 Foto da pulseira Empatica E4, fornecida no site do fabricante. 42
3.10 Diagrama do Ajuste Dinâmico de Dificuldade híbrido. Os sistemas são representados em verde; as etapas relacionadas à performance em vermelho; e à afetividade, em amarelo [1]. 44

3.11 Captura de tela do mesmo segmento com diferentes níveis de dificuldade. (a) Sem aplicação de ADD; (b) ADD de performance aplicado com o valor de dificuldade global 0,1; e (c) com o valor em 0,9 [1]. 47
3.12 Captura de tela do mesmo segmento com as diferentes amplitudes de pulo. (a) Amplitude máxima sem ADD aplicado; (b) Com ADD de pulo aplicado com o valor de dificuldade global 0, ou seja, mais fácil; e (c) Com o valor em 1, ou seja, mais difícil. 50

4.1 Valores médios e desvio padrão das respostas sobre experiência e imersão de todos os participantes na segunda bateria de testes, separados por caso de ADD selecionado. 54
4.2 Valores médios e desvio padrão das respostas sobre a percepção quanto ao pulo da personagem de todos os participantes na segunda bateria de testes, separados por caso de ADD selecionado. 55
4.3 Média das respostas determinando o quanto os espinhos foram desafiadores em sua experiência em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 57
4.4 Média das respostas determinando o quanto os inimigos com ataque a distância foram desafiadores em sua experiência em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 57
4.5 Média das respostas determinando a dificuldade que o jogador teve em controlar o pulo da personagem em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 59
4.6 Média das respostas determinando a satisfação e percepção da responsividade do pulo para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 59
4.7 Média das respostas determinando se o desafio foi justo para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 60
4.8 Média das respostas determinando se a dificuldade foi apropriada para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 60
4.9 Média das respostas determinando o quanto o jogador gostou da sua experiência com o jogo em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários. 61
4.10 Média das dificuldades medidas durante as partidas de cada voluntário por nível, separado por modelo de ADD. 61

Lista de Tabelas

2.1 Exemplos de taxonomia dos jogos eletrônicos (retirada de [1]). 7

4.1 Porcentagem de níveis que foram completados por cada usuário categorizados por tipo de ADD escolhido. 56
4.2 Tempo mínimo, médio, máximo e mediana relacionados a permanência em cada nível da segunda bateria. 58

Capítulo 1

Introdução

1.1 Definição do problema

Ao longo das últimas décadas, a indústria dos jogos eletrônicos experienciou um grande crescimento, se tornando um mercado multibilionário e maior que diversas outras indústrias do entretenimento [6], como a do cinema. Como consequência, os desenvolvedores desses *softwares* passaram a almejar um público mais amplo como consumidor dos seus produtos. Assim, se torna necessário a adequação aos novos cenários, criando jogos divertidos para diversos tipos de perfis de jogador. Faz-se necessária a existência de novos mecanismos que permitam adaptar os jogos aos usuários, assim abrangendo distintos níveis de habilidades e preferências.

Os sistemas adaptativos buscam fornecer ao jogador uma experiência customizada, realizando ajustes no jogo conforme objetivos identificáveis, mensuráveis e influenciáveis [7]. Ou seja, procuram alcançar o equilíbrio do jogo para distintos jogadores de forma consistente, justa e divertida [8]. A atividade, portanto, pode manter o jogador em um estado de fluxo, oferecendo um desafio constante e de acordo com a evolução do jogador, permitindo a ele se manter motivado a continuar jogando com um desafio proporcional ao seu desempenho [9].

A experiência de jogar um jogo não-adaptativo pode ser entediante ou frustrante, o que ocorre por não levar em conta a correspondência entre o grau do desafio proposto e o perfil do jogador [10]. Tendo isso em mente e em contraste aos níveis de dificuldade classicamente estabelecidos (como fácil, médio e difícil), existe o Ajuste Dinâmico de Dificuldade (ADD), uma técnica emergente que busca ajustar a dificuldade de um jogo em tempo de execução de acordo com as habilidades de cada jogador [11].

O Ajuste Dinâmico de Dificuldade é tradicionalmente baseado em dados de desempenho do jogador, isto é, o ajuste é aplicado em elementos específicos de jogo, tais como o número de inimigos, tempo de recarga de uma arma, velocidade dos inimigos, número de

obstáculos, tamanho e número de buracos entre plataformas [1], etc. Porém, no que diz respeito à alteração de variáveis do avatar do jogador utilizando o ADD, os estudos são escassos. Através de pesquisas nas bases de dados da *IEEE Xplore Digital Library*[1], da *Association for Computing Machinery* (ACM)[2], do *Google Scholar* [3] e do *Researche Gate* [4], não foram encontrados trabalhos que tratassem sobre a aplicação de Ajuste Dinâmico de Dificuldade no pulo do avatar em um jogo de plataforma.

1.1.1 Objetivos

Neste trabalho, objetiva-se comparar a aplicação de diferentes casos de aplicação de Ajuste Dinâmico de Dificuldade por desempenho em um jogo de plataforma: (i) sem ADD, (ii) ADD aplicado à plataforma, (iii) ADD aplicado ao pulo e (iv) ADD aplicado tanto ao pulo quanto à plataforma. Verificou-se quais dos elementos, ao serem ajustados, induzem uma melhor experiência e manutenção do estado de fluxo. O estudo foi feito analisando tanto a dificuldade experienciada pelos usuários quanto a medida através de métodos computacionais. O trabalho foi construído em cima do que foi proposto e estudado por Rosa [1].

Utilizando do conteúdo sob o qual essa pesquisa foi desenvolvida, o estudo consistiu de duas baterias de testes, uma piloto e uma definitiva, e apenas uma etapa de análise, feita em cima da segunda bateria. Ambas foram realizadas de forma remota e virtual. A partir dos testes piloto, aferiu-se que o jogo adaptado no presente estudo a partir do trabalho de Rosa estava funcional e que o sistema pensado para receber dados através da internet era íntegro. O teste piloto foi pensado como uma ferramenta iterativa, sendo repetido em caso de falha em uma de suas averiguações.

Com o teste piloto apresentando resultados positivos, foi conduzido o teste definitivo com um total de 155 participantes, divididos pelos quatro casos analisados. Utilizando dados de questionários apresentados ao início e ao final de cada teste e dados do jogo gravados pelo computador, realizou-se a análise conjunta desses dados para comparar a eficiência quanto manter o jogador em estado de fluxo em cada um dos casos de aplicação.

1.1.2 Estruturação

Este trabalho foi estruturado da seguinte forma:

[1] https://ieeexplore.ieee.org/
[2] https://dl.acm.org/
[3] https://scholar.google.com.br/
[4] https://www.researchgate.net/

- **Capítulo 2:** denominado Fundamentação Teórica, define, a partir de trabalhos prévios, os conceitos necessários para a compreensão do assunto abordado nesta pesquisa.
- **Capítulo 3:** descreve o trabalho de Rosa, em cima do qual este estudo é construído, assim como detalha a proposta do trabalho e as soluções empregadas.
- **Capítulo 4:** relata os testes conduzidos e os resultados obtidos, bem como as respectivas análises.
- **Capítulo 5:** por fim, apresenta as conclusões desta investigação e sugestões para a continuidade dos estudos como trabalhos futuros.

Capítulo 2

Fundamentação teórica

Este capítulo apresentará conceitos fundamentais, necessários para o entendimento do trabalho em totalidade. Serão discutidos diversos conceitos, tais como o jogo e sua taxonomia, o jogador e suas classificações, as emoções e o fluxo, além dos tipos de ajuste de dificuldade.

2.1 Jogo

A definição do que é um jogo já foi amplamente discutida por diversos autores, já que impacta uma imensa área de estudos. Em um estudo realizado em 2016, Stenros [12] analisa mais de 60 definições feitas desde 1930, trazendo tal pesquisa como uma ferramenta de valor para estudiosos da área de jogos. Apresentando análises acerca de 10 questões, as quais permitiriam uma melhor definição por cada pesquisador, o autor apresenta análises que contribuem para a compreensão dos desafios para especificar o que é considerado um jogo, diminuindo as divergências sobre um conceito tão fundamental. Ainda assim, há contraste no que seria a conceituação de um jogo por parte dos pesquisadores.

Dentre as definições com maior destaque no meio acadêmico e apesar da discordância entre distintos autores, este trabalho decide utilizar como base a definição trazida por Juul [13]. Segundo Juul, um jogo é um sistema formal baseado em regras com um resultado variável e quantificável, de maneira que resultados diferentes possuem valores distintos, tal que o jogador exerça esforço com o objetivo de influenciar o resultado e que este jogador se sinta conectado emocionalmente ao resultado. Em complemento, as consequências da atividade devem ser opcionais e negociáveis.

Ao melhor definir a interpretação proposta, Juul apresenta 6 pontos relativos ao conceito de jogo:

1. **Regras fixas:** um jogo necessita de regras bem definidas e coerentes dentro de suas próprias conceituações. De tal forma, permite-se que seja implementado de

maneira determinística por um computador que garanta o funcionamento dessas regras ou, alternativamente, de maneira analógica, onde qualquer ambiguidade nas regras levaria a uma parada no jogo até a resolução de tal conflito;

2. **Resultado variável e quantificável:** para algo funcionar como um jogo, as regras devem produzir resultados diferentes. Para funcionar como uma atividade, deve-se estar ajustado ao nível de dificuldade do jogador. Sendo variável, o resultado pode ser diferente a cada vez que uma partida ocorre. Adicionalmente, ao se definir tal resultado como quantificável, garante-se que não há dúvidas em relação ao grau de vitória (ou derrota) do(s) jogador(es). Um exemplo seria um jogo tradicional de *arcade* como *Space Invaders*. A pontuação de partidas diferentes pode ser distinta, assim como podem ser maiores ou menores entre si;

3. **Valor designado a possíveis resultados:** alguns dos possíveis resultados dos jogos são melhores que os outros. Um exemplo seriam jogos que possuem mais de um jogador, onde geralmente estes competem entre si por esses resultados de forma que se estabelece um conflito, gerando uma tensão no jogo. Em complemento, há uma tendência de que os resultados positivos são mais difíceis de alcançar do que os resultados negativos, trazendo o fator desafio ao jogo;

4. **Esforço do jogador:** o esforço do jogador é outra maneira de mostrar que jogos são desafiadores; é esperado que sejam interativos e que ações do jogador influenciem o resultado final, com este fator trazendo uma ligação do jogador ao resultado;

5. **Jogador ligado ao resultado:** a ligação do jogador ao seu resultado é um fator psicológico que significa que esse se torna ligado a aspectos específicos do resultado. Ele pode se sentor feliz quando ganha e triste quando perde. Tal fator normalmente incentiva o jogador a se esforçar mais para alcançar a vitória. Porém, mesmo que o resultado não dependa de tal variável, o mesmo sentimento pode existir. Um exemplo seriam jogos baseados em sorte e aleatoriedade;

6. **Consequências negociáveis:** um jogo é caracterizado pelo fato de poder ser opcionalmente atribuido alguma consequência na vida real, seja em termos de partida, pessoa ou localização. Um exemplo seria a comparação entre um jogo como *The Legend of Zelda* e um caça-níqueis. Enquanto em um é possível e totalmente opcional apostar em seu resultado, no outro a aposta se torna necessária para que possa ser executado.

Os pontos 1, 2 e 4 evidenciam as propriedades do jogo como um sistema formal. O ponto 3, o objetivo que o jogador deve almejar. Os pontos 4 e 5, a relação entre o sistema e o jogador. O ponto 6, a relação entre a atividade de jogador e o resto do mundo.

Especialmente o ponto 5 é importante para o contexto desse trabalho, pois influencia na resposta positiva ou negativa que o jogador pode ter a partir do resultado obtido no jogo.

Um ponto essencial quanto ao jogo é a diversão [14], se relacionando ao esforço do usuário e aos resultados por ele alcançados. Para Koster [15], a diversão nos jogos advém da compreensão e do domínio de padrões, os desafios do jogo. Por sua vez, o tédio surge da necessidade de absorver novos padrões, quando o jogo não é mais capaz de fornecer variações desses ou com a dificuldade não condizente as habilidades do jogador.

Em relação a jogos digitais, Salen e Zimmerman [16] afirmam sobre o meio físico do jogo ser um elemento importante, porém esse não constitui o sistema por inteiro. Assim, sistemas digitais não devem ser enfatizados como um fim em si, mas ser entendidos como um elemento em um sistema maior. Com isso, quatro traços são estipulados para resumir as qualidades especiais de um jogo digital:

1. Interatividade imediata, mas estreita;
2. Manipulação da informação;
3. Sistemas complexos autônomos;
4. Comunicação em rede.

Observa-se que esses traços estão presentes em jogos não digitais, ainda que jogos digitais os encorporem de forma mais robusta. Ademais, as propriedades e desafios do design de um jogo se mantêm verdadeiras independente do meio de manifestação.

Complementar aos conceitos expostos, Salen e Zimmerman [17] definem *gameplay* como um processo reativo, delimitado por regras e emergente de sequências de ações pertencentes a um ou mais jogadores. Esse termo é comumente confundido com jogabilidade(*playability*, que, segundo Xavier [18], é uma decorrência natural do lidar com o jogo, assim como o termo usabilidade é com produtos e processos de manipulação. Ou seja, é uma característica intrínseca ao conjunto de ações que são esperados do jogador com todos os componentes, sejam eles audiovisuais, emocionais ou puramente cognitivos.

Mais especificamente, segundo Prado e Vannucchi [19], o *gameplay* é como um espectro de possíveis ações e reações geradas pelo jogador e pelo jogo, sendo o ponto de encontro desses. Enquanto a jogabilidade pode ser afetada pela entrada (teclado, controle, mouse), o *gameplay* é indiferente nesse ponto. Como exposto por Mello e Perani [20], a jogabilidade diz respeito a como o jogador interage com o jogo e o *gameplay* se associa às mecânicas de jogo, com esse resultado da interação entre a jogabilidade e as regras. No caso de mecânicas de jogo, com esse sendo o resultado da interação entre a jogabilidade e as regras. No caso, as mecânicas de jogo são as regras governando a exploração do espaço de possibilidades do jogo por esse processo e possuem uma grande influência sobre a experiência do jogador.

Tabela 2.1: Exemplos de taxonomia dos jogos eletrônicos (retirada de [1]).

Classificação		**Exemplos**
Quanto ao estilo		
Jogos de ação e habilidade	Luta Tiro Plataforma	Street Fighter, Mortal Kombat, Space Invaders, Quake 3, Super Mario Bros, Castlevania
Jogos de estratégia	Turnos RPG Exploração	Civilization, Diablo, Final Fantasy, Starcraft
Simuladores	Transportes Militares Construção Sociabilização	Flight Simulator, America's Army, Sim City, Zoo Tycoon, The Sims, Second Life
Esportes	Corrida Coletivos Individuais	Pole Position, F1 2010, FIFA Soccer, NBA Jam, Virtua Tennis, Chessmaster
Educativos	Matérias escolares Conhecimento	Coelhinho Sabido, Carmen Sandiego, Show do Milhão
Quanto ao número de jogadores		
Jogador único		Uncharted, Sonic
Múltiplos jogadores		Contra, World of Warcraft

2.1.1 Taxonomia dos jogos

A classificação taxonômica dos jogos revela princípios fundamentais do design de jogos e explicita similaridades e diferenças entre famílias de jogos e entre membros de cada família, conforme apresentado por Crawford [21]. Entrentanto, ele alerta sobre nenhuma classificação ser totalmente correta, em vista da possibilidade de definição de diversas taxonomias por distintos aspectos do jogo. Ademais, com a evolução da área surgem novos jogos que preenchem lacunas de taxonomias prévias e, com isso, geram necessidades de novas taxonomias para refletir essas mudanças.

A Tabela 2.1 mostra algumas taxonomias com base na jogabilidade, no gênero e na quantidade de jogadores. Essa classificação não é absoluta, em vista de vários jogos se encaixarem em mais de uma categoria e haver discordância entre autores, mas serve como um guia para os estudos específicos de gênero.

2.1.2 Categorização de jogos de plataforma

O conceito de plataforma não tem uma definição formal e alguns títulos específicos estão em uma zona onde não está claro se esse rótulo é adequado [5]. Esse tipo de esclarecimento não pretende ser a referência do ponto de vista do jogador, mas fornecer um

quadro de análise e representação do lado científico. Com isso em vista, um conceito genérico apresentado por Minkkinen [22] define que jogos de plataforma tem como cerne um personagem controlado pelo jogador, chamado avatar, que corre e pula para evitar obstáculos e/ou derrotar inimigos.

Para Mourato existem 3 aspectos principais para se analisar jogos de plataforma, baseados nas mecânicas mais comuns [5]:

1. Movimento, que expressa o intervalo de movimentos incluídos no avatar e o respectivo controle sobre esses movimentos;
2. Confronto, que expressa a importância dada a um ambiente de confronto representado com os adversários; e
3. Interação no ambiente, representando funcionalidades de jogabilidade adicionais que não estão diretamente relacionadas com a ideia original do jogo, mas apresentam desafios adicionais e um conjunto de ações mais complexo.

Com base aos aspectos e funcionalidades identificados nas principais propostas encontradas na indústria de jogos, assim , como ilustrado na Figura 2.1, Mourato propôs a definição de jogos de plataforma nos seguintes subgêneros:

- Plataforma clássica: contém os elementos essenciais desse tipo de jogo, concentrando as partes desafiadoras em dominar o movimento do avatar dentro do cenário. Elementos adicionais podem existir, mas com uma influência muito limitada sobre o jogo. Exemplos: *Prince of Persia* e *Sonic the Hedgehog*.
- Plataforma minimalista: reduz a ideia original a personagens menos controlados, com movimento automático e ações instantâneas com pouca estratégia geral. Exemplo: *Canabalt*
- Plataforma de enigmas: estende os princípios da ideia original, permitindo tipos adicionais de ações e interação com o ambiente. Exemplos: *The Lost Vikings* e *Braid*.
- Plataforma de ação: estende o conceito no eixo da confrontação, em que o número de oponentes tende a ser maior. Exemplos: *Super Mario Bros.* e *Rick Dangerous*.
- Plataforma de compostos: complementa a jogabilidade com as características referentes a interconexões e interações de ambiente, sem enfocar nenhum desses aspectos em particular. Assim, cria um mecanismo de jogo mais complexo, mas mantém o equilíbrio entre esses recursos. Exemplo: *Little Big Planet*.

Figura 2.1: Imagens dos jogos *Super Mario Bros.* (A), *Little Big Planet* (B) e *Sonic the Hedgehog* (C) [2].

Alguns jogos não podem ser classificados como um todo, pois foram projetados como híbridos. Caso o conteúdo do jogo seja decomposto em segmentos nos quais a jogabilidade é diferenciada significativamente, consequentemente não se encaixarão em uma única categoria, possuindo apenas os chamados níveis de plataforma dentre de seu domínio [5]. Por exemplo, os jogos *The Adventures of Tintin* e *Another World*, que possuem estrutura de plataforma e outros gêneros (corrida e quebra-cabeça) durante sua narrativa.

2.2 Jogador

A conceituação do termo jogador é importante para compreender o relacionamento dele com o jogo e para alcançar uma melhor experiência de usuário, objetivada neste trabalho. De acordo com Fullerton [23], para se tornar um jogador é preciso aceitar voluntariamente as regras e as restrições do jogo. Em complemento a isso, o jogador é um agente que busca diversão no jogo [14]. Ou seja, o que motiva o jogador a jogar é a busca pela diversão no reconhecimento e domínio de padrões de um jogo [15].

Procura-se entender o que dá prazer ao jogar, o nível de desafio almejado, a motivação de estar jogado, o que se busca no jogo, os componentes do jogo que amplificam a experiência de jogar e quaisquer outros fatores do jogo que podem ser adaptados para melhorar a experiência do jogador [24]. Conforme existem esses distintos fatores, o jogador pode ser classificado de acordo com as motivações para preferir determinados jogos ou possuir comportamentos específicos ao jogar.

2.2.1 Classificação de jogador

Em contextos de entretenimento e para méritos empresariais e acadêmicos, comumente se divivdem os jogadores em grandes categorias, por exemplo por fatores comportamentais, demográficos ou psicológicos [8]. Desse modo, a categorização pode ser feita em base ao nível de dedicação, pelo tipo primário do jogo, por questões culturais e de gênero, por aspectos motivacionais ou por uma combinação desses e demais fatores.

Para Salen e Zimmerman [16], os jogadores podem ser divididos nos seguintes tipos, de acordo com a aderência às regras:

- Jogador casual;
- Jogador dedicado;
- Jogador antiesportivo;
- Jogador trapaceiro;
- Jogador desmancha-prazeres.

Ainda que hajam muitas tentativas de formalização, não há um consenso sobre as definições ou nome dessas categorias, dependendo do autor e do foco de estudo. Para este trabalho, são considerados dois tipos de jogadores, os casuais e os experientes, como é mais adotado pela comunidade de desenvolvedores [8].

Jogadores casuais jogam com menor frequência, preferindo jogos rápidos que não exigem tanta dedicação e proporcionam uma distração rápida, como jogos de celulares ou via *web*. Relacionam-se aos jogadores casuais descritos por [16], jogando conforme foi designado.

Por outro lado, os jogadores experientes empregam bastante tempo em jogos. Ademais, inclinam-se a escolher jogos mais complexos, que possuem narrativas ou mecânicas mais elaboradas. Relacionam-se aos jogadores dedicados de [16], procurando dominar e se aperfeiçoar no jogo, geralmente encontrando e explorando estratégias incomuns para ganhar.

Conforme explicitado por Fortugno [25], os jogadores casuais não abordam o mesmo conjunto de habilidades dos jogadores experientes. Por conseguinte, possuem níveis diferentes de exploração do jogo e de paciência com falhas. No caso, os jogadores dedicados, por estarem mais acostumados a explorar e experimentar, são mais tolerantes a frustrações causadas por falhas e pela dificuldade de assimilar conhecimento. Enquanto isso, os casuais não segue, estas convenções e não estão condicionados a tolerar falhas.

2.3 Fluxo

Quando pensam no tipo de experiência que faz uma vida melhor, a maior parte das pessoas pensam em prazer. Segundo Csikszentmihaly [26], o prazer (*pleasure*) é um sentimento de contentamento o qual pode ser atingido quando alguma informação no consciente diz que expectativas estabelecidas por condições biológicas ou sociais foram atingidas. Por sua vez, o prazer (*enjoyment*, definido diferentemente de *pleasure* por Csikszentmihaly) é caracterizado como o sentimento ao alcançar-se resultados, formente ligado ao que seria uma

conquista. Jogar um jogo desafiador, ler um livro, fechar um contrato de negócios, entre outras situações são exemplos de coisas que são prazerosas (*enjoyable*). É o sentimento de novidade ou realização, sendo descrito como possuindo 8 componentes principais:

1. Tarefas com uma possibilidade de conclusão;
2. Objetivos claros;
3. *Feedback* imediato;
4. Envolvimento profundo, mas sem esforço, que elimina da consciência as frustrações e preocupações da vida cotidiana;
5. Senso de controle sobre nossas ações;
6. Nenhuma preocupação pelo eu;
7. Alteração do conceito de tempo.

Combinando os termos previamente citados, tem-se o conceito de fluxo [26], o qual oferece uma explicação teórica para o prazer. Quando o jogador se encontra em estado de fluxo, a atenção que o indivíduo investe em uma atividade é feita livremente, pois não há desordem para fortalecer ou alguma ameaça da qual se necessite defesa. Dito isso, o fluxo ocorre como um equilíbrio entre as habilidades do invíduo em questão em conjunto com a dificuldade da tarefa proposta.

No fluxo, existem dois processos psicológicos amplos: diferenciação e integração. O eu torna-se diferenciado quando a pessoa, após uma experiência de fluxo, se sente mais capaz e habilidosa, promovendo a individualidade. O fluxo leva a integração porque pensamentos, intenções, sentimentos e sentidos estão focados no mesmo objetivo, facilitando as conexões e a segurança.

Segundo o que Cowley estipula [3], a teoria do fluxo indica a possibilidade de que qualquer pessoa seja capaz de experimentar o fluxo com qualquer atividade. No entanto, é necessário um grande investimento de tempo e esforço para tal, além de que, como pré-requisitos, as habilidades da pessoa sejam compatíveis com os desafios propostos em uma proporção correta e que o indivíduo também tenha uma personalidade autotélica (ou seja, capaz de reconhecer e aproveitar oportunidades de fluxo).

Considerando o isomorfismo do fluxo [26], no entanto, faz-se necessário a existência de atividades universalmente acessíveis que antecipam e capacitam estados cognitivos universais e inerentes à experiência do fluxo [3]. Dessa forma, Cowley indica que algumas atividades são mais sucetíveis ao fluxo por qualquer pessoa independente de sua habilidade, como é o caso dos jogos. Com a entrada do usuário no estado de fluxo, ocorrem diversos efeitos. Dentre eles, um aumento do auto-controle, concentração na atividade em

questão, uma clara visão dos objetivos, perda de autoconsciência, sensação alterada do tempo, um aumento na motivação pessoal e também uma combinação de ação e percepção.

Corroborando os estudos de Csikszentmihalyi e os associando à mídia, constatou-se [27] que os aspectos do fluxo são mantidos para mídias nos formatos tradicional e também interativo, tais como televisão e jogos eletrônicos, respectivamente. Dessa forma, a capacidade de um indíviduo de atingir o estado de fluxo através de uma determinada mensagem de mídia está ligada a como o entretenimento é experienciado por ele. Em consequência, ao entender-se tais mecanismos subjacentes ao prazer e à preferência de mídia, tem-se um facilitador para melhores designs de mídias pró-sociais. Ao mesmo tempo, o engajamento é facilitado ao associar-se com o estado de fluxo, prolongando o uso do material.

2.3.1 Fluxo em jogos

Algo em comum entre jogos de sucesso é direcionar o foco do jogador para o jogo, fazendo-o se abstrair do mundo real para dentro da realidade do jogo e provendo uma distração do mundo real. Uma pessoa pode chegar a não perceber o tempo passando e nem os acontecimentos ao seu redor por estar focada quase que totalmente no jogo, resultando no jogador se sentindo como se fosse o próprio personagem que controla. Ao atingir esse estado, o jogador está imerso no jogo [3].

A imersão é um fator crítico para o aproveitamento do jogo e uma boa experiência do jogador. Tal estado pode ser atingido ao proporcionar uma atividade cujo o desafio seja suficiente para que não se perca o interesse por tédio e que nem supere suas habilidades e se sinta frustrado. As Figuras 2.2 e 2.3 ilustram, respectivamente, a correlação entre tais variáveis com o fluxo e outras experiências.

Os videogames como mídia podem ser analisados como dois componentes essenciais [28]:

- **Conteúdo do jogo:** seria a experiência específica a qual o jogo é projetado para transmitir.
- **Sistema do jogo:** um software interativo capaz de comunicar o conteúdo do jogo para seus usuários através de visuais, aúdio e interações.

Chen [28] analisa que, ao ser visto como conteúdo, a definição de fluxo é muito ampla, necessitando de conteúdos mais sofisticados para casos específicos. Visto como um sistema, tem-se uma explicação do porque as pessoas tendem a preferir ou viciar mais em certos jogos. Caso o jogo alcance todos os elementos principais do fluxo, qualquer conteúdo poderá ser recompensador e sua premissa engajadora [29].

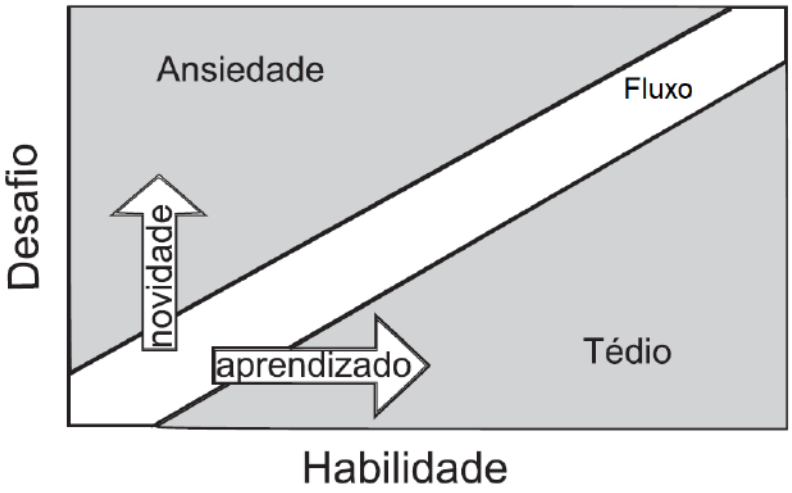

Figura 2.2: Diagrama do canal de fluxo, adaptado por Rosa [1], contendo as três dimensões de uma experiência: ansiedade, fluxo e tédio [3].

Figura 2.3: As oito dimensões da experiência [4]. Consiste no diagrama do modelo de flutuação da experiência tal que o ponto médio do gráfico corresponde a média de habilidades e desafios de uma pessoa para uma dada atividade [3].

Segundo a Figura 2.4, cada jogador possui uma curva de experiência adquirida ao jogar uma parcela de um jogo, se mantendo em uma zona segura. No caso da experiência se distanciar da zona de fluxo, a entropia psíquica negativa (ansiedade, tédio) irá interromper o estado de fluxo. Além disso, diferentes indivíduos possuem diferentes habilidades e zonas

Figura 2.4: a) Experiência do estado de fluxo em um jogador; b) Jogador ao encontrar entropias psíquicas; c) Diferentes jogadores e zonas de fluxo (retirado de [1]).

de fluxo. Ou seja, um jogo bem projetado é capaz de manter jogadores casuais no fluxo, porém não fará o mesmo com tal eficácia em relação a jogadores experientes e novatos. Gera-se uma relação entre a habilidade do usuário e o desafio para que este se mantenha no estado de fluxo.

Para criar um jogo para públicos diversos, tem-se então que a experiência de jogo não pode ser linear e estática. Pelo contrário, necessita oferecer uma ampla cobertura de possíveis experiências para então inserir-se nas zonas de fluxo de jogadores diferentes. Para tal equilíbrio ser atingido, pode-se realizar um ajuste de desafio, mantendo a relação entre desafio e habilidade proporcional.

2.4 A dificuldade em jogos

A função de um jogo eletrônico, como já enunciado, seria de forma geral a diversão [14]. Porém, não há uma fórmula pré-definida para alcançá-la pois, assim como a dificuldade, depende de diversos elementos essenciais. Dimensionar a dificuldade é algo fundamental para o projeto de um jogo, o que ocorre através da configuração de obstáculos a serem superados voluntariamente [30]. Portanto, um bom *game design* necessita de uma curva de aprendizado e dificuldade coerentes [31].

A dificuldade é derivada da relação entre habilidade e desafio, ou seja, quanto maior for a habilidade necessária para resolver um desafio, maior será sua respectiva dificuldade [4]. Dessa maneira, um jogo é entediante quando fácil demais e frustrante quando muito difícil. É notável como, ainda assim, o sentimento subjetivo da dificuldade é complexo, não estando ligado somente ao resultado final [30].

Sabendo que a dificuldade é um elemento subjetivo que varia de jogador a jogador, dependendo de sua habilidade, existem maneiras de ajustar o grau de dificuldade oferecido, influenciando o conteúdo apresentado e inclusive alterando regras de jogo. Tradicionalmente, existe a separação interna em jogos por níveis de dificuldade, tais como fácil, médio

e difícil. Todavia, há uma crescente demanda para a adaptatividade em jogos, como o Ajuste Dinâmico de Dificuldade.

Olhando para os primórdios do desenvolvimento de jogos, encontra-se casos de jogos que não usavam o conceito de dificuldade. Alguns exemplos são *SpaceWar!*, *Pong* e *Gun-Fight*. Isso ocorria devido à simplicidade no *design* desses jogos, que possuíam desafios únicos e requisitos de habilidade constante [24]. Outros jogos fazem uma escala do desafio, aumentando gradativamente a cada fase (*Mario*, *Mega Man*) ou a cada nível da personagem (*Elder Scrolls*). Além disso, alguns jogos adaptam o desafio à habilidade do jogador (*Mario Kart*), possuem opções acerca do foco do jogador na narrativa ou na ação (*Divinity: Original Sin 2*) ou sugerem mudança de nível no caso de muitas derrotas consecutivas (*God of War*).

De modo informal, Nicollet [32] apresentou uma visão geral sobre padrões de projeto em relação a dificuldade. Apesar de alguns princípios serem discutíveis, este é um bom ponto de partida para examinar o assunto [5]. A partir disso, algumas regras importantes são estipuladas para representar a dificuldade:

- Surpresa não é o mesmo que dificuldade. Portanto, eventos aleatórios sem padrão definido não tornam um desafio mais difícil pois, de tal maneira, não pode ser dominado. Um exemplo seriam certos elementos de jogos *roguelike*, onde os níveis aleatórios tornam impossível o domínio do jogador sob esses.

- Dificuldade implica em falha possível e provável. Por sua vez, a falha implica na perda de algo, sendo que quanto maior essa perda, maior a percepção da dificuldade. Um exemplo seria, novamente, jogos *roguelike*, pois nestes a derrota do jogador implica em ter de recomeçar seu progresso. Este é popularmente conhecido como um dos fatores que traz maior dificuldade a jogos do gênero.

- Reduzir as janelas de tempo de uma ação ou um conjunto de ações aumenta a dificuldade. Jogos de ritmo, como *Osu!* são um bom exemplo pois, para gerarem dificuldade, esses jogos exigem cada vez mais entradas por parte do jogador em espaços de tempo cada vez menores.

- Uma sequência contínua de ações tem maior dificuldade do que as mesmas ações realizadas de forma independente. Pode-se observar tal efeito em jogos de luta como *Street Fighter*, onde para obter melhores ataques é necessário executar uma sequência certa de ataques e movimentos, o que é mais difícil que executar cada uma dessas ações individualmente.

- Reduzir o controle do jogador sobre o personagem geralmente aumenta a dificuldade.

- Aumentar a precisão para superar um desafio aumenta a dificuldade em relação a esse desafio em questão. Isso pode ser observado em alguns jogos da série *Mario*, onde existem elementos de cenário (como ventos fortes ou chãos de gelo escorregadios) que influenciam no movimento tradicional do jogador, aumentando a necessidade do jogador ser preciso nos seus movimentos.

2.4.1 Adaptatividade

Um sistema adaptativo é composto por várias entidades interativas ou interdependentes, reais ou abstratas, que formam um todo integrado capaz de responder as mudanças físicas ou ambientais nas partes que interagem [33]. Ao ser aplicado, tal sistema tem por objetivo realizar ajustes com a função de melhorar alguma característica do usuário.

Empresas desenvolvedoras de jogos, em busca de uma melhor experiência em seus produtos, gastam tempo e recursos com pesquisas e testes para encontrar dificuldades ideais em jogos através de versões Alfa e Beta (as quais podem ser fechadas para teste interno ou abertas para o público geral). Entretanto, tais testes ainda podem falhar na busca de um balanceamento ideal [34].

Dentre as possíveis soluções, uma seria a customização da experiência. Esta seria a criação de uma experiência diferente para cada jogador através de sistemas adaptativos. Quanto mais se sabe sobre o jogador e sua habilidade por meio do desempenho, mais fácil encaixá-lo em algum perfil determinado previamente e então fazer ajustes mínimos para corresponder a expectativa do jogador, assim almejando mantê-lo no estado de fluxo.

Visando atingir um estado determinado por Novak [8] como sendo um jogo equilibrado, um sistema adaptativo pode ser aplicado neste. Um requisito para o jogador entrar em estado de fluxo seria o jogo estar equilibrado e apresentar as seguintes características:

1. Desafios balanceados;
2. Experiência justa;
3. Um jogo em que o jogador consiga avançar;
4. As decisões dos jogadores podem influenciar a história do jogo;
5. A possibilidade de escolher um nível de dificuldade adequado.

Classificam-se jogos, então, em três tipos [4]:

1. **Não adaptativo:** é o jogo que não almeja adequar sua dificuldade ao jogador, independente de desempenho. Podem ter uma seleção de níveis ao início, ou então possuir uma dificuldade que aumenta enquanto se progride no jogo. São exemplos jogos como os das franquias *Mario* e *The Legend of Zelda*;

2. **Semi-adaptativo:** é o jogo que possui seleção de nível (assim como em alguns casos de jogos adaptativos) e, adicionalmente, possuem algoritmos que registram o desempenho do jogador continuamente, realizando comparações quanto ao nível registrado e a dificuldade de jogo almejada. Em casos do jogador estar com baixo desempenho, o próprio jogo poderia oferecer a opção de reduzir seu nível de dificuldade, tal como acontece no título *God of War*;

3. **Jogo adaptativo:** em um jogo adaptativo, a seleção de nível inicial pode ou não existir. De maneira parecida com os semi-adaptativos, este realiza o registro do desempenho do jogador constantemente. Porém, ao perceber um baixo ou alto desempenho, o jogo é capaz de ajustar o nível de dificuldade para atingir o desafio ideal. O jogo *Half Life 2* possui um exemplo de adaptatividade, onde a quantidade de suprimentos de vida que aparecem ao jogador são baseados na vida atual do jogador.

2.4.2 Ajuste dinâmico de dificuldade

Ajuste dinâmico de dificuldade (ou ADD) é um método que permite a modificação automática de característas, comportamentos e cenários de um jogo durante sua execução. Essas alterações podem ser feitas com base na habilidade do jogador de maneira que este não se sinta entediado ou frustrado no caso do jogo ser muito fácil ou muito difícil, respectivamente. O objetivo do ADD é, afinal, manter o jogador imerso durante todo o jogo, proporcionando-lhe uma experiência desafiadora. Como já dito anteriormente, jogos tradicionais tendem a possuir níveis de dificuldade que aumentam linearmente ou gradualmente durante o dercorrer do jogo. Recursos, tais como munição ou quantidade de vidas, podem ser estabelecidos somente no início ao se definir um nível de dificuldade. Isso pode, no entanto, resultar em uma experiência negativa, o que ocorre devido a dificilmente o jogador se enquadrar exatamente em um dos níveis de dificuldade disponibilizados (como, por exemplo, fácil, médio ou difícil). Através do ADD, busca-se uma solução personalizada para o problema apresentado [35].

Sabendo que o ADD busca fazer um contraponto aos métodos tradicionais de escolha de dificuldade, assim permitindo um melhor balanceamento da dificuldade, tem-se esse como uma técnica que ajusta automaticamente elementos de jogos (tais como cenários, parâmetros ou comportamento de inimigos) [36]. Um sistema de ajuste dinâmico de dificuldade deve suprir os três pontos a seguir:

- O jogo necessita ser capaz de registrar rapidamente a habilidade do jogador, respondendo e se adaptando a ela;

- O jogo deve estar de acordo com a melhora ou piora da habilidade do jogador, mantendo o balanceamento de acordo com o desempenho apresentado;
- A atuação do sistema de adaptação não deve ser perceptível ao jogador. Além disso, os níveis posteriores tem de manter alguma progressão de dificuldade em relação aos níveis anteriores.

Existem diversos métodos de ADD. Muitos usam de variáveis presentes no estado atual do jogo, tais como pontuação, taxa de acertos ou tempo de jogo sem progredir, para determinar valores para certas configurações de jogo. Por meio desse método, calculam uma nota para o desempenho do jogador. Classifica-se os sistemas de ajuste dinâmico de dificuldade, segundo Zohaib [35], da seguinte maneira:

- **Métodos probabilísticos:** tal aproximação visa maximizar o engajamento do jogador ao longo do jogo. Modela-se a progressão do jogador em um grafo probabilístico que maximiza o engajamento como uma função objetiva e bem definida. Um jogo mais complexo que possuí progressões múltiplas ou não-lineares gera grafos mais complexos, pois necessita de mais estados e conexões. Alguns exemplos de técnicas utilizadas seriam algoritmos genéticos, modelos ocultos de Markov e otimização Bayesiana.
- ***Perceptron* simples e multicamadas:** por meio do mapeamento de características percebidas no comportamento do jogador, parâmetros de *design* de nível e até mesmo emoções do jogador, *perceptrons*, o tipo mais simples de rede neural, podem ser treinados visando o ADD. Em alguns trabalhos, são utilizados para definir a função de desempenho ou então para gerar níveis aleatórios de acordo com a dificuldade do jogador.
- **Algoritmos dinâmicos:** usando de técnicas de aprendizado não supervisionado, é um método rápido, robusto, eficiente e efetivo. Nesse método, o aprendizado acontece progressivamente. Ao completar-se um encontro, os pesos das regras utilizados no encontro são analisados dependendo do seu efeito no resultado, fazendo com que os pesos de regras que levaram ao sucesso sejam maiores e daquelas que levaram ao fracasso sejam menores.
- **Sistema de *Hamlet*:** muitos jogos utilizam o conceito de inventário, ou seja, um estoque de itens que o jogador coleta e armazena ao longo do jogo. A ausência ou abundância de itens no inventário afeta diretamente a experiência do jogador, sendo que jogos são projetados para manter o controle sobre a troca de itens entre jogador e mundo. Em essência, o sistema de Hamlet visa modelar tais conexões produtor-consumidor como um sistema dinâmico ou até mesmo como uma economia,

ajustando oferta e demanda de itens de maneira a manipular a dificuldade de jogo. Caracteriza-se por:

1. Gerenciar estatísticas do jogo de acordo com as métricas previstas;
2. Decidir quais seriam as tarefas de ajuste e regras;
3. Lidar com tais tarefas e regras;
4. Mostrar os dados e as configurações do sistema.

Logo, os sistemas de Hamlet tem como funcionalidade prever situações em que o jogador não teria condições de passar, assim fornecendo itens o suficiente para tal desafio ser superado, ajustando assim a dificuldade.

- **Aprendizado reforçado:** sabendo que, devido a variedade de jogadores que usam as mais diversas estratégias e padrões de jogo, tem-se que uma IA estática de jogo não pode lidar com os estilos de jogo de todos os jogadores. Logo, visa-se uma IA adaptativa, assim criando experiências variadas para diferentes estilos de jogo. Algoritmos evolucionários vem sendo uma forma de aplicar esse conceito na literatura.

2.4.3 Dificuldade em jogos de plataforma

Alteração de dificuldade é algo particular ao gênero de cada jogo, resultando em implementações específicas e variadas. Tal característica pode ser percebida ao comparar três casos que utilizavam do conceito de Ajuste Dinâmico de Dificuldade (ADD): uma adaptação de *Half-Life* (um FPS 3D) [37], uma do jogo *Super Mario Bros.* (plataforma 2D) [38] e uma do jogo *Tetris* (*arcade* 2D) [39]. Na adaptação para o FPS, usou-se de economia de itens para gerar o ADD. No de plataforma, um cálculo sob a porcentagem de moedas coletadas em comparação a vidas e *power ups*. Para o arcade, por sua vez, utilizou-se do histórico do jogador para melhorar o fornecimento de peças.

É perceptível como, em cada um dos casos previamente citados, o ADD foi construído com base em elementos intrínsecos não somente ao gênero adaptado, mas às vezes a elementos de cada jogo em si. Portanto, não pode ser algo replicado entre eles. Visando analisar a dificuldade em jogos de plataforma, é importante estabelecer o que constituí um jogo desse gênero, indo de seus elementos fundamentais até a maneira ocorre a combinação destes de maneira a criar-se um nível de jogo. Determina-se tais elementos como os componentes, padrões e estruturas celulares [40].

Componentes são as unidades básicas que constituem a composição de um jogo de plataformas, tais como vinhas, plataformas, colinas e picos. Podem ser tanto um obstáculo quanto local de descanso, podendo possuir diversas propriedades físicas, tais como

potencial de causar dano ou serem escorregadios. Visando categorizar componentes pelo seu propósito no nível, temos os seguintes [41]:

- **Plataformas:** elementos que permitem ao jogador pular sobre ou usar para atravessar o nível;
- **Obstáculos:** qualquer objeto que cause dano ou a morte do avatar, como espinhos, inimigos ou lacunas (explicitando que o obstáculo não necessita ser explicitamente um objeto no nível);
- **Ajudas de movimento:** elementos que concedam ao jogador ajuda para atravessar o nível. São exemplos molas, trampolins ou cordas;
- **Itens colecionáveis:** objetos que forcenem ao jogador alguma recompensa. Moedas, poderes e outras recompensas pontuais são possíveis exemplos;
- **Gatilhos:** objetos que, de alguma forma, alteram o estado do nível. Possíveis exemplos incluem chaves que transformam blocos em moedas ou elementos que alterem o comportamento do avatar ou de outra regra de jogo.

Já os padrões, por sua vez, fornecem o mecanismo para agrupar componentes individuais em uma sequência mais longa ao mesmo tempo que mantém um movimento rítmico para o jogador [40]. Existem quatro tipos de padrões:

- **Padrão básico:** consiste em um componente sozinho ou então repetido várias vezes sem variação.
- **Padrão complexo:** é uma repetição de um mesmo componente, porém com ajustes tais que fiquem de acordo com alguma sequência definida, como, por exemplo, uma série de saltos horizontais de largura crescente.
- **Padrão composto (*compound*):** alterna entre os padrões básicos feitos de dois tipos diferentes de componentes. Um exemplo seria uma sequência de um número qualquer de saltos horizontais, seguido por outro número qualquer de espinhos que, por sua vez, são seguidos por mais outro número de saltos horizontais novamente.
- **Padrão composto (*composite*):** consiste em dois diferentes componentes colocados tão próximos um do outro que passam a requerer um tipo diferente de ação ou então uma ação coordenada, tal que esse seja um movimento que não seria necessário para cada um individualmente. Através disso, o jogador se vê tendo que levar em conta ambos os componentes para então elaborar uma solução. Um exemplo seria um desafio que combine duas mecânicas, como uma lacuna seguida por uma

plataforma e logo após outra lacuna (o que faz com que o jogador tenha que controlar a distância do pulo) em conjunto com um obstáculo cronometrado que cause dano ao jogador enquanto paira por cima da lacuna (fazendo com que a contagem de tempo se torne um fator também).

Através dos padrões, pode-se construir sequências estritamente lineares. Ainda assim, existem elementos não lineares, os quais capacitam o jogador a escolher suas próprias trajetórias no jogo. Logo, define-se células como blocos de construção não lineares, feitas a partir do encapsulamento de padrões lineares [40]. Ou seja, formam uma estrutura de árvore hierárquica que representa uma relação entre entre os componentes, onde estes seriam as folhas e o nível completo a raíz. Dessa forma, pode-se focar em dividir o nível em áreas de desafio:

1. A maior subdivisão de um nível é chamada de célula. Esta é uma seção de jogabilidade linear, as quais são conectadas por portais;
2. As células, por sua vez, são compostas de grupos rítmicos, os quais são conjuntos não sobrepostos dos componentes do nível. Tais grupos são geralmente pequenos, encapsulando seções desafiadoras do jogo. Intervalos entre grupos rítmicos são locais seguros para o jogador descansar antes de prosseguir com o nível.

Sorenson [42] analisa os padrões de *design* de jogos tais como *Super Mario Bros.*, definindo um sistema genérico baseado em uma implementação de um algoritmo genético cujo o objetivo é gerar níveis através desses padrões.

- **Bloco:** caracterizado por suas coordenadas x e y;
- **Tubo:** caracterizado pela sua coordenada horizontal x, sua altura e uma possível presença de inimigo;
- **Lacuna:** caracterizada por uma coordenada horizontal x e um valor de largura;
- **Escadaria:** caracterizada por uma coordenada horizontal x, possibilidade de ser ascendente ou descendente, e também um valor de largura que pode ser visto como um número de degraus;
- **Plataforma:** caracterizada por uma coordenada horizontal x e uma altura definida y, tendo um valor adicionar pra sua largura;
- **Inimigo:** caracterizado por uma certa coordenada horizontal x, já que o exemplo usado possui apenas um típo de oponente por conveniência.

2.5 A experiência do jogador

Durante a execução de uma atividade otimizada, possibilita-se o surgimento de uma sensação de prazer, a qual não ocorre em momentos passivos e sim quando o corpo ou mente é colocado em seu limite em um esforço voluntário para realizar algo difícil ou satisfatório [26]. Na mente, tais experiências são representadas como informação e estados ótimos são resultados da ordem na consciência. Logo, ao combinar um foco em objetivos realistas em relação a habilidades individuais com oportunidades de ação, o indivíduo pode se concentrar na tarefa em questão.

As abordagens utilizadas para maximizar a satisfação do jogador podem ser classificadas nas categorias implícitas e explícitas. A primeira consiste na alteração de parâmetros periféricos à satisfação do jogador, ou seja, com um foco em variáveis de jogo, por exemplo. As explícitas referem-se a maximização de uma função que representa a satisfação do jogador. Para otimizar a experiência do jogador e, em consequência, maximizar sua satisfação, este trabalho fará uso de um mecanismo de Ajuste Dinâmico de Dificuldade desenvolvido por Rosa [1]. Logo, a abordagem usada será implícita e não haverá uma função que define a satisfação.

Capítulo 3

Proposta de ADD em característica da personagem

A presente pesquisa é uma continuação de uma das linhas de trabalhos futuros propostos pela monografia de graduação de Rosa [1]. Portanto, explicar sua pesquisa é essencial para a compreensão do que o atual trabalho almeja alcançar. Neste capítulo, as partes relevantes do trabalho desenvolvido por Rosa serão apresentadas, seguidas da proposta de modificação da atual pesquisa, assim como as ferramentas, o projeto e os equipamentos utilizados para a realização de testes e para a coleta de dados.

3.1 Trabalho de Rosa

O objetivo do trabalho de Rosa [1], no qual esta pesquisa se baseia, era investigar um método para estimar a dificuldade de sessões de jogos de plataforma e desenvolver um mecanismo híbrido de Ajuste Dinâmico de Dificuldade. Esse mecanismo deveria se adequar ao jogador e mantê-lo em estado de fluxo, comparando sua eficiência com os modelos sem ADD, por desempenho e afetivo. Com a proposta definida, foi produzido um jogo que se adaptasse tanto aos dados extraídos por algoritmos de análise de desempenho do jogador quanto aos dados obtidos por um sensor de captura de Atividade Eletrodérmica (EDA), permitindo estimar a frustação e o tédio do jogador.

Cada participante jogou versões do jogo com distintos modelos de ADD aplicados, além de responder questionários e ter seus dados recolhidos para averiguação dos objetos de pesquisa. O modelo híbrido proposto provou-se capaz de estabilizar o jogador em seu estado de fluxo e de se ajustar às habilidades dele com resultados superiores aos demais modelos estudados.

3.1.1 O Jogo

Requisitos

Considerando que o propósito desta pesquisa não é o desenvolvimento de um jogo eletrônico, o foco foi na adaptação do projeto original utilizado na pesquisa de Rosa [1] para ser ajustado às necessidades do estudo.

O jogo em questão deveria seguir todas os seguintes requisitos de validade para o atual trabalho:

- Licença de usos não comerciais e código fonte disponível para alterações;
- Estrutura com componentes independentes para fácil adição de código que avaliasse e armazenasse dados referentes ao desempenho do jogador, e para aplicação do Ajuste Dinâmico de Dificuldade;
- Elementos comuns ao gênero de plataforma, baseado nos estudos de Minkkinen [22], e e conformidade com os padrões de um jogo do subgênero plataforma clássica [2];
- Controles e interações simples para facilitar a ambientação de jogadores menos experientes;
- Jogo para um único jogador, com o propósito de manter o escopo da pesquisa na análise individual do comportamento do jogador somente em relação ao ambiente do jogo;
- Rodar em sistema operacional Windows, com o propósito de padronizar a plataforma de teste, diminuindo possíveis problemas exclusivos a outros sistemas operacionais.

Como esses requisitos continuam os mesmos e nenhum deles foi invalidado desde o estudo anterior, o mesmo projeto *The Explorer: 2D*, ilustrado na Figura 3.1, foi escolhido para a presente pesquisa.

De uso livre, esse jogo foi desenvolvido pela Unity Technologies e contém um conjunto de sistemas, ferramentas, funcionalidades e elementos de jogabilidade para ensinar desenvolvedores de jogos a utilizar seu motor. A desenvolvedora resume o jogo como[1]: "Explore o antigo e misterioso planeta alienígena onde Ellen, nossa engenheira principal, pousou. Desvie dos perigos e derrote os inimigos que espreitam nas antigas ruínas desta civilização alienígena."

[1] `https://assetstore.unity.com/packages/essentials/tutorial-projects/2d-game-kit-107098`

Figura 3.1: Captura de tela da versão original do jogo *The Explorer: 2D*.

Plataforma de Desenvolvimento do Jogo

A *Unity*[2] é uma plataforma de desenvolvimento de jogos 2D e 3D, criada pela empresa *Unity Technologies*. Consiste em um programa de computador que produz abstrações para o desenvolvimento de jogos eletrônicos.

Assim como outras plataformas de desenvolvimento de jogos eletrônicos comerciais, disponibiliza funcionalidades de renderização gráfica, simulações de física, animações, sons, inteligência artifical, gerenciamento de memória, entre outras. É uma plataforma versátil, que permite a criação de versões do jogo para múltiplas plataformas, como computadores, consoles, dispositivos móveis e navegadores web.

No contexto do trabalho de Rosa [1] e da atual pesquisa, o projeto foi desenvolvido para o sistema operacional Windows. A plataforma Unity possui uma quantidade enorme de usuários, além de fornecer suporte online recorrente por meio de fóruns e tutoriais oficiais. Além disso, a empresa disponibiliza uma loja[3] com uma ampla variedade de conteúdo, fornecendo bibliotecas e *assets* de desenvolvimento de jogos 2D e 3D de sua autoria ou de terceiros.

A ferramenta apresentada tem como objetivo simplificar o desenvolvimento de jogos, ao possuir um editor intuitivo e ao permitir que os objetos não sejam limitados aos módulos incorporados à aplicação, adequando-se a uma linguagem de programação orientada a

[2] `https://unity.com/pt`
[3] `https://assetstore.unity.com/`

objetos. A única linguagem de programação aceita atualmente pela plataforma é o *CSharp* (C#), sendo utilizada para desenvolver todas as alterações de ambos os trabalhos.

Versão Original

A primeira versão do jogo é datada de dezembro de 2017 e foi implementada na plataforma *Unity* 2017.3.0. O jogo recebe constantes atualizações para correções de erros e adaptações para novas versões.

Semelhante ao jogo *Super Mario Bros.*, o jogador assume o controle de uma personagem, a exploradora Ellen, que pode se movimentar para frente, para trás e pular, com o objetivo de seguir um caminho linear, desviar de obstáculos e finalizar o percurso. A câmera mantém a Ellen como seu foco, se movimentando conforme sua posição, mas também pode ser controlada para cima ou para baixo pelos controles. A exploradora também possui um conjunto básico de ações: agachar, sacar espada e atirar.

A personagem em questão possui parâmetros variáveis como velocidade, aceleração e desaceleração, redução de velocidade enquanto empurra um objeto, dano recebido e executado, quantidade de vidas, além da posição e velocidade da câmera que a acompanha. Os fatores citados podem se relacionar em ambos os eixos, horizontal e vertical, como andar e pular ao mesmo tempo.

Os comandos padrão para realizar as ações da personagem são:

- Movimentação Lateral: Teclas A e D;
- Salto: Teclas Espaço ou W;
- Agachamento: Tecla S;
- Saque de Espada: Tecla K;
- Tiro: Tecla O.

O usuário pode escolher uma nova configuração para os controles por meio do editor sem o jogo estar inicializado. Durante o jogo, é permitido apenas pausar para ajuste de volume, visualização da configuração atual dos comandos ou encerramento do jogo. A tela de pausa é semelhante ao menu principal, que possui as funções mencionadas e a de inicialização do jogo.

No total, o projeto possui 5 fases prontas e o desenvolvimento de novas fases é liberado e encorajado pela empresa Unity Technologies, utilizando sua plataforma de desenvolvimento de jogos. O jogo possui um sistema de vidas onde, ao se alcançar um número limite de mortes, o jogador é teletransportado para o início da fase, perdendo seu progresso até

então realizado. As fases prontas possuem inimigos de alcances diferentes e caminhos não-lineares, sendo necessário resolver quebra-cabeças como localizar chaves ou se posicionar em cima de painéis de pressão para abrir portas. A Figura 3.2 demonstra tanto a Tela de Alerta, também conhecida como *Heads-up display* ou HUD, quanto a interface de sistema de diálogo. A HUD contém informações visuais de importantes para o jogador, como o número de vidas restantes e porcentagem de chaves coletadas. A interface de sistema de diálogo contém texto com finalidade de mostrar as falas da personagem, seja para a história do jogo ou para explicar comandos ao jogador de uma maneira mais orgânica.

Figura 3.2: HUD contendo vidas, itens coletáveis e interface do sistema de diálogo da versão original do jogo *The Explorer: 2D*.

Adaptações do trabalho de Rosa

De acordo com Rosa [1], o projeto já possuia estruturação, entretanto foi necessário realizar diversas mudanças para adequá-lo à proposta de seu trabalho. Desenvolveu-se um gerador de fases usando imagens ou arquivos json, evitando utilizar as ferramentas da plataforma *Unity* para tal. O jogo foi adaptado para cada fase possuir um padrão de grade, ou seja, ladrilhos quadrados de dimensão fixa. Com isso, facilitou-se a organização dos elementos da fase de modo padronizado e as modificações dos componentes durante a sessão de jogo, pois o posicionamento e comportamento das plataformas possuem uma referência numérica inteira.

As fases pré-existentes e a interface de sistema de diálogo foram descartadas, pois seu estudo necessitava de níveis criados com arquivos de entrada para executar as alterações em tempo real e não inclui diálogos que precisassem da interface descartada.

Em seu estudo, as fases são construídas em uma cena vazia, conforme o carregamento de novos arquivos de entrada. Cada unidade de pixel representa um componente na fase e a cor do pixel identifica o tipo de elemento ou descrição de comportamento no caso das plataformas, conforme representado na Figura 3.3.

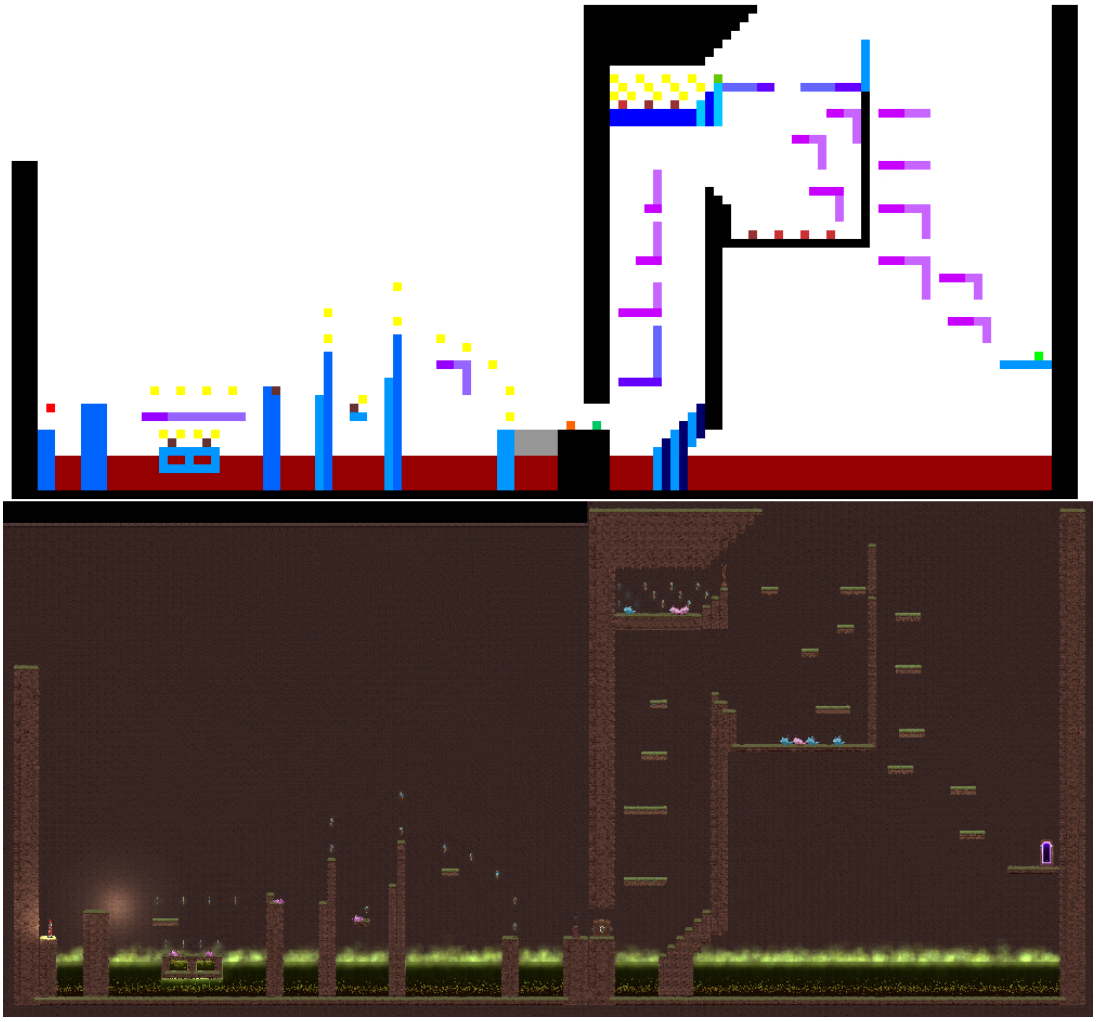

Figura 3.3: Arquivo de entrada para a geração da fase e o resultado visual, adaptada de [1].

Todos os componenetes que seriam úteis à pesquisa de Rosa [1] foram categorizados, conforme a Figura 3.4, e atribuídos a cores diferentes. São esses:

- Personagem jogável, a exploradora Ellen;

- Chaves, elementos colecionáveis do jogo, semelhante em função às moedas de *Super Mario Bros.*;
- Componentes de plataforma:
  - Plataforma padrão que ao colidir repele o jogador e possui posição estática;
  - Plataforma móvel, semelhante à padrão, porém com padrão de movimentação variável horizontal e/ou vertical;
  - Plataforma atravessável que não repele o jogador ao colidir, deixando passar pela parte inferior ao saltar e possuindo a opção de cair através dela; e
  - Abismo, objeto que não possui solo para a personagem se apoiar, caindo até causar a morte da personagem.
- Componentes gerais:
  - Pedra e pilar destrutíveis que quebram ao serem acertados por um saque de espada;
  - Caixa de madeira que pode ser empurrada horizontalmente;
  - *Checkpoints*, pilares visíveis ou invisíveis que indicam o progresso do jogador em um certo trecho da fase e servem para posicionar o retorno da personagem ao morrer; e
  - Portal, objeto que transporta a personagem para outra fase.
- Componentes mortais:
  - Blocos invisíveis que causam a morte da personagem ao colidir e se encontram na parte inferior do mapa para quando o jogador cai em um abismo ou sai da região da fase;
  - Ácidos, sendo um líquido verde translúcido que causa morte ao colidir; e
  - Espinhos, objetos pontiagudos e rosas no qual o jogador morre ao colidir.
- Inimigos:
  - *Chomper*, movimenta-se em direção ao jogador e executa um ataque corpo-a-corpo; e
  - *Spitter*, mantém-se em uma posição fixa e atira projéteis em direção ao jogador.

Na pesquisa de Rosa [1], o enfoque foi na movimentação, ao invés do confronto contra os inimigos ou da interação com o ambiente. Portanto, as prioridades são plataformas

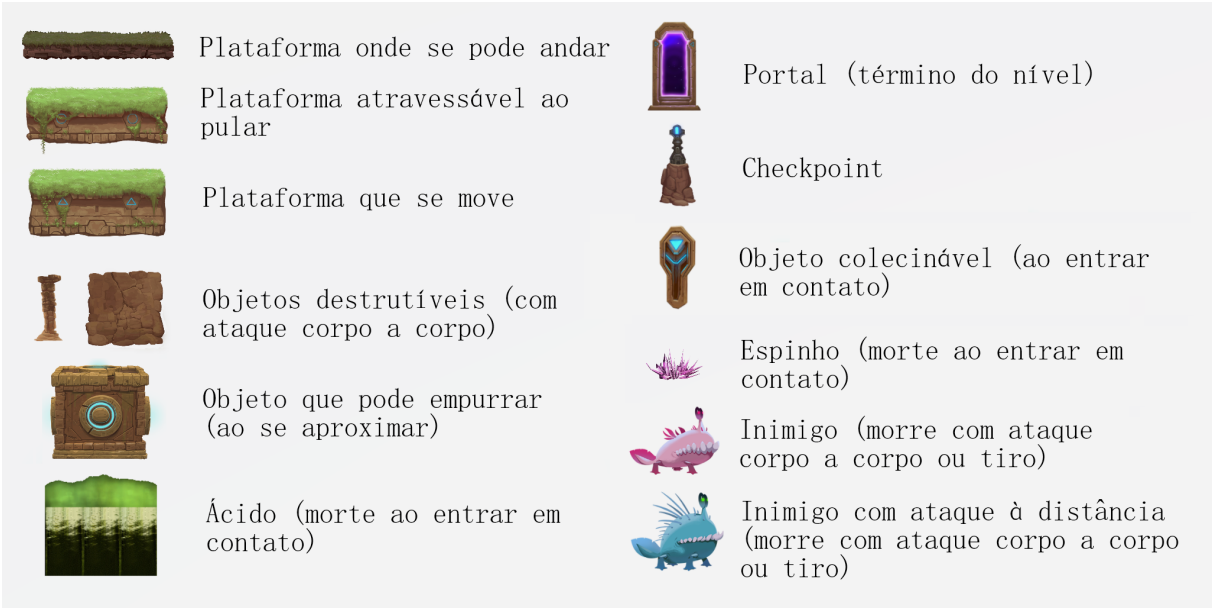

Figura 3.4: Principais componentes categorizados do jogo, conforme ilustrado ao usuário durante a explicação dos comandos no início do jogo [1].

e obstáculos ao se criar fases pela natureza mais determinista em comparação a outros componentes [43].

Ao gerar uma fase, todos os elementos do mapa são divididos por segmento e tipo e são salvos para consultas posteriores, servindo para estimação da dificuldade e ajuste dinâmico. Além disso, manteve-se uma coerência no posicionamento dos componentes para manter a imersão do jogador, principalmente quando próximos e do mesmo tipo. Também foi escolhido um fundo padrão para ocupar a região válida da fase e suas bordas, mantendo consistência visual sem afetar os outros elementos.

Todos as fases possuem a personagem jogável e um portal de saída. Ao gerar a fase, cria-se automaticamente blocos invisíveis na parte inferior do mapa, que causam morte caso a Ellen caia no abismo ou escape da região válida da fase. Uma fase é composta por segmentos, melhorando performance computacional e permitindo alterações frequentes no ajuste dinâmico. Caso o segmento não seja o último da fase, o seu fim é determinado por um *checkpoint*, representado visualmente por um pilar.

Ao iniciar o jogo, uma tela de carregamento é mostrada ao jogador enquanto os dois primeiros segmentos da fase são criados a partir de um arquivo de entrada. O próximo segmento é carregado ao se passar de um *checkpoint*, até que seja o último segmento. Sendo o último segmento, o *checkpoint* se transforma em um portal de saída, levando o jogador à próxima fase. Nessa troca de níveis, o projeto de Rosa [1] poderia aplicar um questionário para analisar o estado de fluxo do jogador. Ao encerrar a último fase, o jogo

apresenta um último questionário e uma tela de agradecimento.

Para que o jogador saiba como controlar a personagem e os desafios que irá enfrentar, foram acrescentadas interfaces de apresentação com os componentes principais e os comandos do jogo no início da sessão e no menu de pausa. Antes do jogador iniciar a primeira fase, um questionário sociodemográfico foi adicionado, com respostas textuais, múltipla escolha, escala, grade de escalas e caixa de seleção.

Tanto o menu principal quanto o menu de pausa foram alterados, acrescentando novas opções de desistência de uma fase e troca de comandos para executar as ações da personagem. É possível apenas escolher entre duas configurações pré-definidas:

- Movimentação Lateral: Teclas A e D ou Teclas Seta para Esquerda e Seta para Direita;
- Salto: Teclas Espaço e W ou Teclas Espaço e Seta para Cima;
- Agachamento: Tecla S ou Tecla Seta para Baixo;
- Saque de Espada: Tecla K ou Tecla S; e
- Tiro: Tecla O ou Tecla Q.

O limite de vidas da versão original foi removido, portanto a personagem possui vidas infinitas, podendo morrer quantas vezes quiser. Essa decisão reduz o risco de afetar drasticamente a reação dos usuários, principalmente no quesito frustração. Portanto, a contagem de chaves coletadas foi acrescentada e o limite de vidas foi removido da HUD.

3.1.2 Coleta de dados do trabalho de Rosa

Os dados para medir o desempenho do usuário e efetuar o Ajuste Dinâmico de Dificuldade foram extraídos tanto do jogo quanto de um sensor de Atividade Eletrodérmica. Mesmo que não fossem utilizados para o ADD, vários dados foram coletados para verificar quais mais se correlacionam com o desempenho dos usuários:

- Dados temporais:
    - Tempo para finalizar segmento;
    - Tempo para finalizar fase;
    - Tempo de cada vida da personagem;
    - Tempo para finalizar segmento;
    - Tempo de movimentação (esquerda, direita ou parado); e
    - Tempo de cada ação da personagem (pular, agachar, sacar espada e atirar).

- Dados quantitativos:
    - Número de mortes;
    - Número de chaves coletadas;
    - Número de sucessos e falhas em saques de espada;
    - Número de sucessos, falhas e ações randômicas em pulos; e
    - Número de interações com cada componente da fase.
- Outras informações:
    - Causas de cada morte; e
    - Distância caminhada em cada vida.

Com os dados listados acima, são realizadas proporções com valores anteriores e referente ao total de um segmento, de uma fase ou de uma vida. Alguns dos dados extraídos de uma sessão de jogo não são medidas diretas, ou seja, é necessário que um processamento seja feito para determiná-los. Analisando o caso das mortes, é preciso averiguar com qual componente a personagem interagiu ao morrer, sendo possível ser a colisão com um conjunto de espinhos, com os inimigos ou seus ataques, com um abismo ou com um ácido. Ao colidir, a posição da personagem é salva e armazenada, sendo registrada como a posição da morte. Para o objetivo do estudo de Rosa [1], interagir com os componentes citados causa morte ao invés de causar unidades de dano.

Para determinar o sucesso ou falha de qualquer ataque, verifica-se a morte do inimigo, se o mesmo colidiu com um saque de espada ou um tiro. Caso seja um desses casos, o ataque é considerado como um sucesso. Senão, o ataque é considerado uma falha ou uma ação randômica executada pelo usuário.

Para determinar o sucesso ou falha de um pulo, é necessário determinar os objetivos do jogador para criar uma lógica que possa estabelecer uma relação com suas ações:

- Objetivo principal: percorrer a fase até encontrar um portal de saída; e
- Objetivo secundário: coletar chaves.

Com as metas estabelecidas, foi determinado os tipos de plataforma conforme a origem do pulo, ou seja, com base na origem prevista da personagem em relação à posição da plataforma:

1. Esquerda Direita;
2. Inferior Esquerda;

3. Inferior Direita;
4. Superior Esquerda; e
5. Superior Direita.

Plataformas definidas como Superior aludem à origem mais elevada do pulo, onde o trecho final do arco encontra-se em uma posição abaixo da posição inicial do pulo. Plataformas definidas como Inferior são o inverso, tendo uma origem mais baixa que o destino do pulo. Plataformas Esquerda aludem à origem do pulo na parte esquerda da mesma e plataformas Direita, na parte direita. Ademais, plataformas Esquerda Direita se associa aos buracos ou abismos, onde a personagem pode pular de qualquer origem horizontal, sem variação vertical entre a origem e o destino do pulo. Uma representação visual foi ilustrada na Figura 3.5.

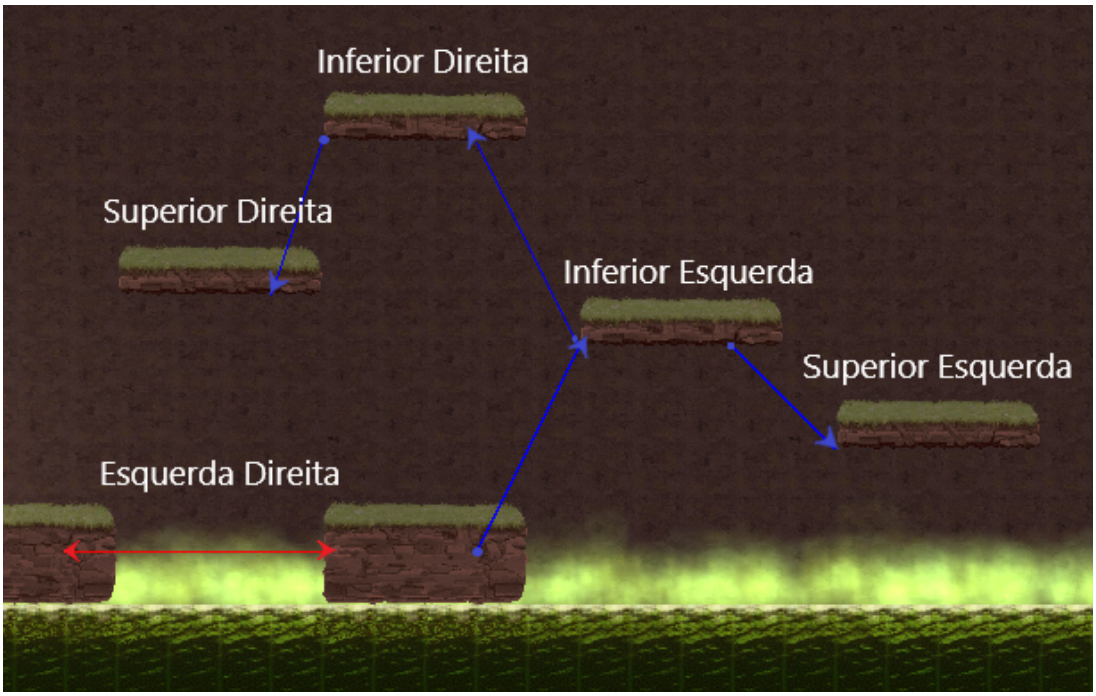

Figura 3.5: Tipos de plataformas. As setas apontam para a plataforma destino do pulo, com seu nome correspondente associado. O caso de exceção Esquerda Direita tem setas apontando para ambas as direções e se refere a um buraco ou abismo no lugar de uma plataforma [1].

Conforme as decisões anteriores, organizou-se por prioridade as características de um pulo de sucesso, falho ou randômico:

1. Morte por queda em abismo ou contato com inimigo ou obstáculo mortal é considerado uma falha;

2. Troca de plataformas, onde a personagem colide com um componente que não seja a plataforma de origem do pulo:

    a. Colisão durante o pulo sem ser com os pés:

        i. Se a plataforma final fica abaixo da origem e for dos tipos de busca Superior Esquerdo ou Superior Direito, o pulo é considerado um sucesso;

        ii. Se a plataforma final não é a mesma do início, não é dos tipos de busca Superior Esquerdo ou Superior Direito e a posição final fica abaixo da origem, o pulo é considerado uma falha:

            · Ao pular de uma plataforma para outra, personagem cai para outra abaixo do destino.

        iii. Os pés colidem com outra plataforma, o pulo é considerado um sucesso; e

        iv. Se a plataforma final é a mesma de início e a distância no eixo vertical entre origem e destino é menor do que o espaço de um ladrilho, o pulo é considerado uma falha:

            · Ao pular de uma plataforma para outra, não alcança com os pés, mas com outra parte do corpo da personagem.

    b. Se a personagem alcança uma parte da mesma plataforma com diferença vertical maior que o espaço de um ladrilho, o pulo é considerado um sucesso; e

    c. Se a personagem atravessou um abismo ou buraco, o pulo é considerado um sucesso:

        * Se a altura final for menor que a altura máxima do pulo, levando em consideração caminhos alternativos que ficam acima de partes da fase que tenham abismos ou buracos.

3. Coleta de chave é considerado um pulo de sucesso; e

4. Nenhuma das opções anteriores é considerado um pulo randômico.

Com todas as características listadas, o trabalho de Rosa [1] as utilizou para determinar todos os pulos de um usuário dentro das três categorias, dando importância às metas previamente apontadas como objetivo para essa classificação. Medir o desempenho do jogador enquanto avança nas fases é possível com esses dados.

3.1.3 Medição da dificuldade do ADD do trabalho de Rosa

A pesquisa de Rosa [1] se propôs a implementar uma versão do método de Mourato [5] para avaliar a dificuldade em uma fase de um jogo do gênero plataforma. Os estudos de

Mourato analisam a geração de fases e sua adaptação para criar estruturas mais complexas e não-lineares, como portões e alavancas. Quanto ao conceito de dificuldade, aplica a ideia de Aponte [30], que considera a relação com as possibilidades de sucesso e falha durante as ações do usuário, sejam elas isoladas ou persistentes por uma fase completa. Com isso, baseia-se em dois tipos de análise:

1. A fase é separada em segmentos que são trechos independentes e, por consequência, são identificadas transições entre esses trechos; e
2. Uma análise de cada componente individual é feita tendo como referência a sua probabilidade de sucesso.

Segundo ilustrado na Figura 3.6, o cálculo da dificuldade do desafio é separado em trechos mapeados:

a) Trecho sem risco de falha, em outras palavras, sem plataformas, abismos, inimigos ou obstáculos mortais;

b) Desafio com uma certa dificuldade, possibilidade de falha e morte, com presença de abismos, inimigos e/ou obstáculos mortais, porém sem plataformas; e

c) Desafio repetível com penalidade:

    1) Possibilidade de falha, porém existe possibilidade de repetição, com presença de plataformas, porém sem chance de morte; e
    2) Possibilidade de desafio adicional, além de chance de falha e possibilidade de repetição e/ou morte, com presença de plataformas, abismos, inimigos e/ou obstáculos mortais.

Reconhecer as possibilidades de transições de segmento em uma fase tendo como base o conteúdo e as características da personagem não é trivial, pois são variáveis para jogos diferentes, com movimentações e elementos diversificados. Com esse modelo de medição, é proposto examinar informações gerais com o contexto de espaço e tempo.

Na questão da dificuldade espacial, uma simulação do pulo é realizada para definir o mínimo que pode ser performado, onde considera dois pontos (origem e destino) e faz o lançamento de um projétil da origem para o destino para medir uma possível margem de erro da trajetória utilizando a interseção.

Na questão da dificuldade temporal, os seguintes elementos são levados em consideração: período da entidade desafiadora, período de prejuízo e tempo necessário para superar a entidade. É definida uma falha quando o usuário colide com a entidade no período de

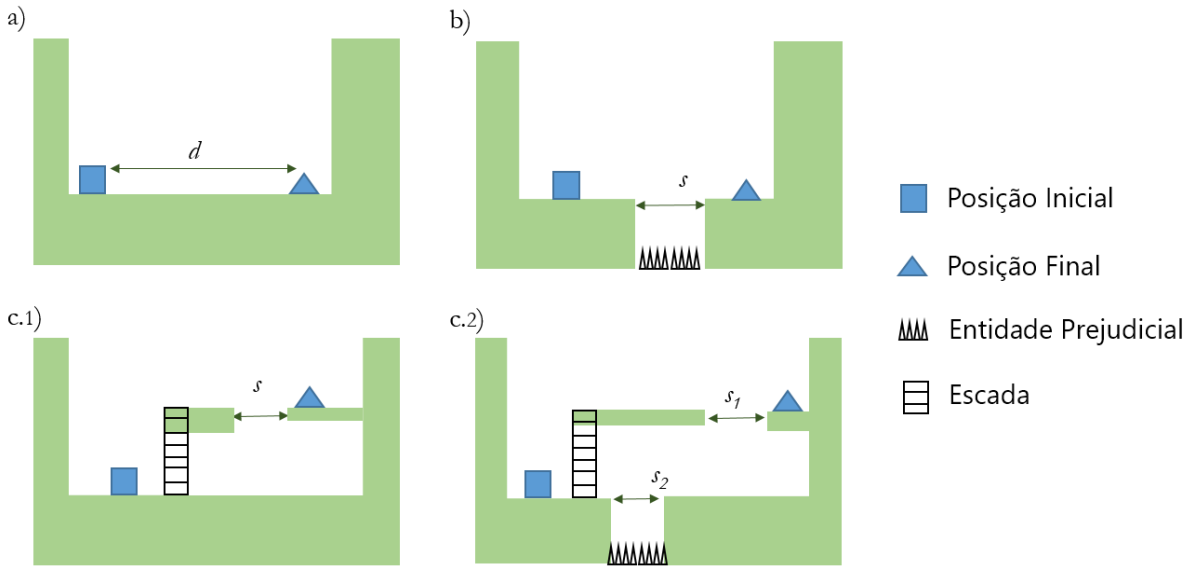

Figura 3.6: Demonstração de desafios em jogos do gênero de plataforma [1].

prejuízo ou se sua movimentação for tardia e não se encaixar no intervalo de superação da entidade.

Combinando as duas, a dificuldade espaço-temporal é medida pela multiplicação das probabilidades de sucesso das situações previamente citadas. São definidas por entidades dinâmicas, como, por exemplo, plataformas móveis.

Rosa realizou modificações nos cálculos para incluir a influência do eixo horizontal na função de tempo da posição no eixo vertical, novos componentes independentes como plataformas atravessáveis e abismo, e a lógica de sucesso do pulo.

Em conclusão, a medição da dificuldade do estudo de Rosa foi fundamentada na lógica de sucesso do pulo. Portanto, foram descartadas como triviais as ações de movimentação e interação com componentes que não causam morte. Entretanto, pulo entre plataformas e abismos ou buracos com sucesso é a característica que irá definir a dificuldade relativa. O pulo representa uma parábola e a possibilidade de sucesso deriva do erro horizontal e vertical da parábola referente a um determinado ponto final.

Implementação da estimativa da dificuldade do trabalho de Rosa

Com o intuito de definir a dificuldade de uma fase e estabelecer o algoritmo que representa o ADD do estudo de Rosa [1], são somadas as dificuldades de componentes que possuem o pulo como foco principal de seu desafio:

- Plataformas básicas;
- Plataformas móveis;

- Plataformas atravessáveis; e
- Abismos ou buracos.

Conforme os conceitos estabelecidos na Subseção 3.1.2, calcula-se uma simulação de lançamento de projétil da origem (P_0) a um ponto intermediário (P_1) e avaliando a margem de erro da trajetória até ao destino (P_2) aplicando a interseção. A origem (P_0) é definida pelo algoritmo de Busca em Largura (BFS), atado pela direção da origem da personagem. Vale ressaltar que a escolha e utilização do BFS para esta pesquisa se deve por encontrar o menor caminho partindo de uma origem em termos de número de arestas no caminho. De acordo com a direção da origem, o ponto intermediário (P_1) é o primeiro contato possível para a personagem.

Para cada caso, a busca é realizada de maneira distinta:

- Inferior Esquerda: BFS é aplicada nos pontos à esquerda e abaixo da plataforma. Origem à esquerda, com a personagem indo para direita e o destino do pulo é para um ponto igual ou mais alto que a origem. P_1 é o ponto mais baixo e à esquerda desse componente.
- Inferior Direita: BFS é aplicada nos pontos à direita e abaixo da plataforma. Origem à direita, com a personagem indo para esquerda e o destino do pulo é para um ponto igual ou mais alto que a origem. P_1 é o ponto mais baixo e à direita desse componente.
- Superior Esquerda: BFS é aplicada nos pontos à esquerda e acima da plataforma. Origem à esquerda, com a personagem indo para direita e o destino do pulo é para um ponto igual ou mais baixo que a origem. P_1 é o ponto mais alto e à esquerda desse componente.
- Superior Direita: BFS é aplicada nos pontos à direita e acima da plataforma. Origem à direita, com a personagem indo para esquerda e o destino do pulo é para um ponto igual ou mais baixo que a origem. P_1 é o ponto mais alto e à direita desse componente.
- Esquerda Direita: O caso excepcional, onde a personagem pode pular de ambos lados. Origem e destino são os pontos à esquerda e à direita mais próximos, respectivamente.

Observa-se que todos os casos acima são ilustrados na Figura 3.5, que se encontra na Subseção 3.1.2, com as setas azuis e vermelhas demonstrando a trajetória da origem

ao destino. No caso de plataformas móveis, considera-se o caminho traçado como posições válidas. No caso de plataformas atravessáveis, a busca continua com um padrão semelhante, acrescentando apenas a procura de áreas diretamente abaixo delas.

No quesito espacial de um componenete, a trajetória do projétil é considerada em função do tempo para representar o avatar e sua movimentação. Portanto, é possível definir as variáveis para as equações:

- x_0 sendo a velocidade inicial;
- v_0 sendo a velocidade inicial no eixo correspondente;
- y sendo a posição atual no eixo vertical; e
- $x(t)$ e $y(t)$ sendo as funções no tempo para os eixos horizontal e vertical, respectivamente.

Com isso, conclui-se as equações:

$$x(t) = x_0 - v_0 * t \tag{3.1}$$

$$y(t) = y_0 + v_0 * y * t + \frac{a}{2} * t^2 \tag{3.2}$$

Para o caso avaliado, a velocidade horizontal é constante e não é levado em consideração a aceleração e a desaceleração no eixo horizontal. Com essas definições estabelecidas e considerando g como aceleração gravitacional do jogo, a seguinte equação é obtida para o eixo vertical:

$$y(t) = y_0 + v_0 * y * t + \frac{g}{2} * t^2 \tag{3.3}$$

Nos cálculos, os seguintes fatores são considerados:

1. Origem é determinada como o primeiro ponto;
2. Não existe um ângulo de lançamento;
3. Velocidade constante, no caso a velocidade máxima alcançável; e
4. Apenas se trata da questão espacial.

Com isso, é obtida uma função quadrática que intersecciona a origem, demonstrada na Equação 3.4, tendo como variáveis:

- v_{0_x} sendo a velocidade máxima do eixo horizontal;
- v_{0_y} sendo a velocidade do pulo;

- x sendo a posição no eixo horizontal; e
- P_0 sendo a origem ($(x_0, y_0) = (0, 0)$).

$$f(x) = \frac{v_{0_y}}{v_{0_x}} * x - \frac{g}{2 * {v_{0_x}}^2} * x^2 \tag{3.4}$$

Conforme ilustrado na Figura 3.7, é possível estimar o desvio da trajetória de um pulo ainda válido, conhecido como margem de erro, interseccionando o projétil com duas retas paralelas ao eixo cartesiano e com o destino. As Equações 3.5 e 3.6 descrevem a estimativa para o eixo x e y, respectivamente. Com isso, é possível observar dois valores:

- Altura da personagem ao alcançar a plataforma em relação à origem Δy; e
- Amplitude horizontal do pulo Δx.

Portanto, determina-se que existe um erro horizontal m_x e vertical m_y, com $\mathrm{P_i} = x_i, y_i$.

$$m_x = \Delta x - x_1 \tag{3.5}$$

$$m_y = \Delta y - y_1 \tag{3.6}$$

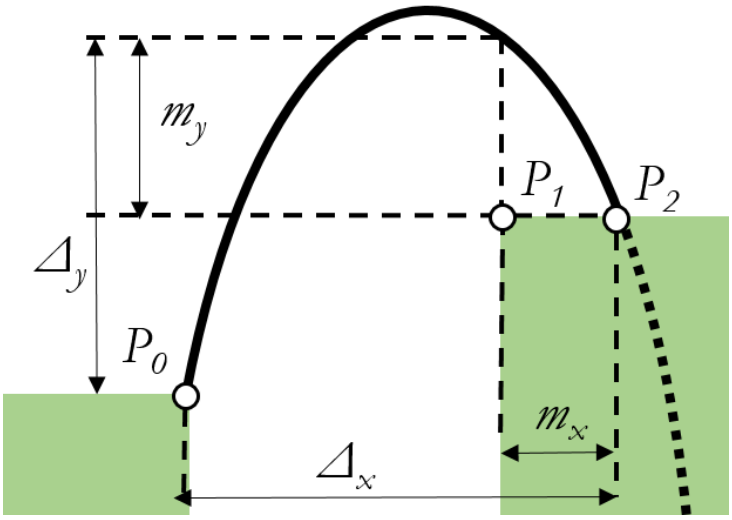

Figura 3.7: Conceito de margem de erro demonstrado visualmente [5].

Em caso os valores de erro sejam negativos, novas posições são calculadas, pois um erro menor ou igual à zero representa um destino inalcançável. Em caso de valores de erro positivos, a tolerância horizontal utiliza o valor de x para $f(x)$ igual a y_1 em uma equação de segundo grau e escolhendo a raiz apropriada. Com isso, são encontradas as

equações necessárias para calcular as margens de erro, uma para o o eixo vertical de P_1 (y_1) e outra para o de P_2 (x_2):

$$y_1 = f(x_1) = \frac{v_{0_y}}{v_{0_x}} * x - \frac{g}{2 * {v_{0_x}}^2} * {x_1}^2 \tag{3.7}$$

$$x_2 = (v_{0_y} \pm \sqrt{{v_{0_y}}^2 - 2 * g * y_2}) * \frac{v_{0_x}}{g} \tag{3.8}$$

Para efetuar a normalização, as margens de erro são tidas como valores relativos à distância total ao decorrer do eixo, distância total no eixo horizontal x_r) e ponto máximo no eixo vertical (y_v). Por fim, as margens de erro normalizadas (M_x e M_y) passam por uma operação para determinar uma margem de erro final (M) para o componente, tendo como resultado um valor entre 0 e 1, conforme demonstrado nas seguintes equações:

$$x_r = \frac{2 * v_{0_y} * v_{0_x}}{g} \tag{3.9}$$

$$y_v = \frac{{v_{0_y}}^2}{2 * g} \tag{3.10}$$

$$M_x = \frac{m_x}{x_r} \tag{3.11}$$

$$M_y = \frac{m_y}{y_v} \tag{3.12}$$

$$M = M_x * M_y \tag{3.13}$$

Em algumas ocasiões, o pulo precisa ser executado previamente no eixo horizontal para atingir uma altura específica. Nesse caso, é calculado a média entre a distância mínima (c_{min}) e a máxima (c_{max}) de correção no eixo horizontal, conforme demonstrado na Figura 3.8. As plataformas atravessáveis não fazem parte do grupo de componentes no qual esse cálculo é aplicado. Ao definir $limite_{min}(x)$ como o limite inicial e $limite_{max}(x)$ como o final da plataforma no eixo horizontal, as seguintes equações são obtidas:

$$c_{min} = (limite_{min}(x) < x_1) \ ? \ (x_1 - limite_{min}(x)) : 0 \tag{3.14}$$

$$\begin{aligned} c_{max} = ((limite_{min}(x) < x_2) \wedge (limite_{max}(x) < x_2)) \ ? \\ (x_2 - limite_{min}(x)) : (limite_{max}(x) - limite_{min}(x)) \end{aligned} \tag{3.15}$$

Considerando que os usuários não pulam no limite da margem final e com o problema previamente apresentado, é desejada uma distribuição centrada em X_d, onde d é a distância até o final da origem ou do alinhamento dela com o destino no eixo x. Portanto, é preciso descobrir o valor máximo de d (d_{max}) para o sucesso do pulo, retirado da curva

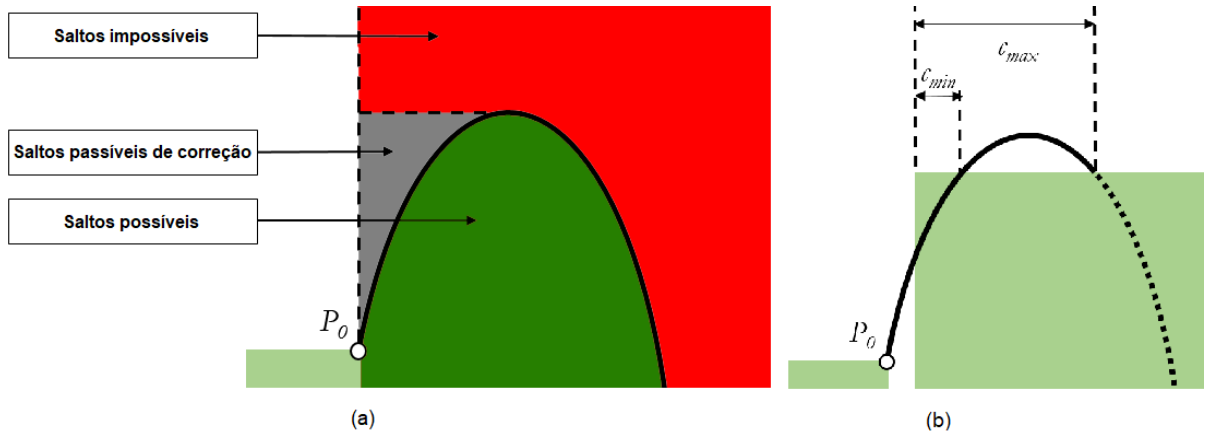

Figura 3.8: (a) Categorização do pulo conforme sua posição de destino (P_1) referente à origem (P_0); (b) Valores de correção para ajustar o pulo de uma trajetória [1].

da trajetória. Nos casos de correção de pulo, a área interna da distribuição do sucesso do pulo é determinada como c_{min} e (d_{max}).

Na situação das plataformas móveis, que possuem tanto características de dificuldade espacial quanto temporal, é feita a multiplicação dos sucessos e usando o valor inverso para estabilizar o valor em uma escala crescente, pois varia entre 0 e 1. No caso da dificuldade espacial, é definida a dificuldade média dos casos de sucesso em que a plataforma esteve presente no trajeto em algum ponto. No caso da dificuldade temporal, é definido o período de tempo necessário para a execução do pulo, adquirido pela associação da velocidade do componente e do espaço onde é incapaz de haver um pulo de sucesso, ou seja, as posições válidas de pulo dividido pela soma de todas as posições.

3.1.4 Sensor do ADD afetivo do trabalho de Rosa

Conforme dito na introdução deste capítulo, apenas as partes essenciais para o presente trabalho seriam detalhadas. Este novo estudo não inclui a utilização ou modificações do Ajuste Dinâmico de Dificuldade afetivo de Rosa [1], assim como sua implementação do ADD híbrido.

Para uma breve contextualização, o sensor de captura de Atividade Eletrodérmica (EDA) foi utilizado para verificar a condutância da pele no trabalho de Rosa [1]. Com isso, foi possível formar uma base de identificação de níveis de excitação, tensão e frustração, aumentando a imersão do jogador.

A pulseira *Empatica E4*[4], ilustrada na Figura 3.9, foi escolhida para aumentar a aceitação e o conforto dos jogadores, mantendo a precisão dos dados coletados, visto que

[4] `https://www.empatica.com/en-int/research/e4/`

aparatos maiores podem ser desconfortáveis para os usuários e afetarem a imersão. Esse aparato é classificado como um dispositivo de pesquisa vestível, que adquire dados fisiológicos em tempo real e possui ferramentas para análise e visualização.

Figura 3.9: Foto da pulseira Empatica E4, fornecida no site do fabricante.

A pulseira *E4* permite a captura de condutividade elétrica pela inversão da resistência por meio da pele. Uma quantidade mínima de corrente elétrica é transferida entre dois eletrodos em contato direto com a pele para prover os dados em unidade de condutância *microSiemens* (μS). O estudo utilizou o relógio interno, marcação de eventos e o EDA para a coleta de dados.

A *Empatica Inc.* fornece aplicativos que permitem o fluxo de dados em tempo real, possibilitando a coleta de dados usando uma conexão Bluetooth com um dispositivo móvel. Os dados brutos foram armazenados em um servidor de banco de dados desenvolvido em PHP e SQL, com a ordenação por chave, valor e data da captura. Também permite o acesso por requisições REST de leitura, inserção, alteração e destruição.

O projeto foi adaptado para coletar os dados recentes ordenadamente em intervalos de 2 segundos. Ademais, os dados foram salvos ao finalizar a sessão, possibilitando a conversão para outros formatos, permitindo análise posterior. Por fim, o jogo processa os dados brutos do EDA para normalizar e inserir no algoritmo de ajuste de dificuldade.

3.1.5 ADD híbrido do trabalho de Rosa

O projeto de Rosa possuia um Ajuste Dinâmico de Dificuldade de performance, que alterava os elementos conforme o desempenho do usuário; um afetivo, que alterava pelo estado afetivo relativo a Atividade Eletrodérmica; e um híbrido, que integra as opções anteriores. Para os três modelos citados, a alteração de dificuldade é feita da mesma forma, ou seja, alteram a mesma variável. Para os dois primeiros casos, existem duas valores globais que representam o desempenho do usuário e seu estado afetivo, variando

de 0 a 1. O terceiro caso é calculado utilizando a média dos dois primeiros. No cálculo do ADD, é utilizada somente uma dificuldade global, dependendo do modelo.

Em resumo, as variáveis do usuário são recolhidas e divididas entre dados brutos do jogo e medidas fisiológicas obtidas pelo sensor citado na Subseção 3.1.4. Então, os dados são filtrados, possibilitando a análise do desempenho e da emoção correspondente. Observa-se que podem ser utilizados de modo independente nos modelos de desempenho e afetivo, respectivamente, ou de modo conjunto no modelo híbrido. A análise ocorre em todos os casos, resultando na definição do que será adaptado nos elementos do jogo.

No caso do ADD afetivo, utiliza-se uma proporção do nível fásico e do nível tônico coletados pelo sensor. A cada 2 segundos, a proporção é recalculada e sua variação é somada à dificuldade global.

No caso do ADD de performance, quando a morte ocorre por ácido ou bloco invisível mortal, é incrementado um contador e a posição na qual a personagem morreu é armazenada. Caso a morte tenha sido em uma posição prévia ou igual a da morte anterior, a dificuldade global do jogo diminui seguindo a proporção estabelecida entre a distância atual e a anterior, assim facilitando a fase. Caso contrário, a dificuldade global aumenta em menor escala, assim dificultando a fase.

A dificuldade aumenta de forma exponencial conforme mais segmentos da fase são finalizados em mortes, e diminui do mesmo modo caso haja mortes, utilizando o contador de mortes mencionado anteriormente. Essas alterações também são aplicadas quando o jogador passa para o portal que troca as fases e é quando ocorre a maior variação da dificuldade global. Por último, é realizada uma proporção e normalização dos números de pulos de sucesso e fracassados, dificultando em casos de alta incidência de sucesso e facilitando quando as falhas são mais frequentes.

Como demonstrado nas Equações 3.16 e 3.17, o valor de dificuldade global é limitado conforme o progresso das fases para todos os casos. Assim, as seguintes variáveis são definidas:

- dif como dificuldade global, que varia entre 0 e 1;
- $maxLevelDif$ como dificuldade limite da fase;
- $level$ como fase atual; e
- $numLevels$ como quantidade de fases.

Exemplificando seu funcionamento, no caso de 12 fases:

- A primeira varia entre 0 e 0,83;
- A sexta varia entre 0 e 1;

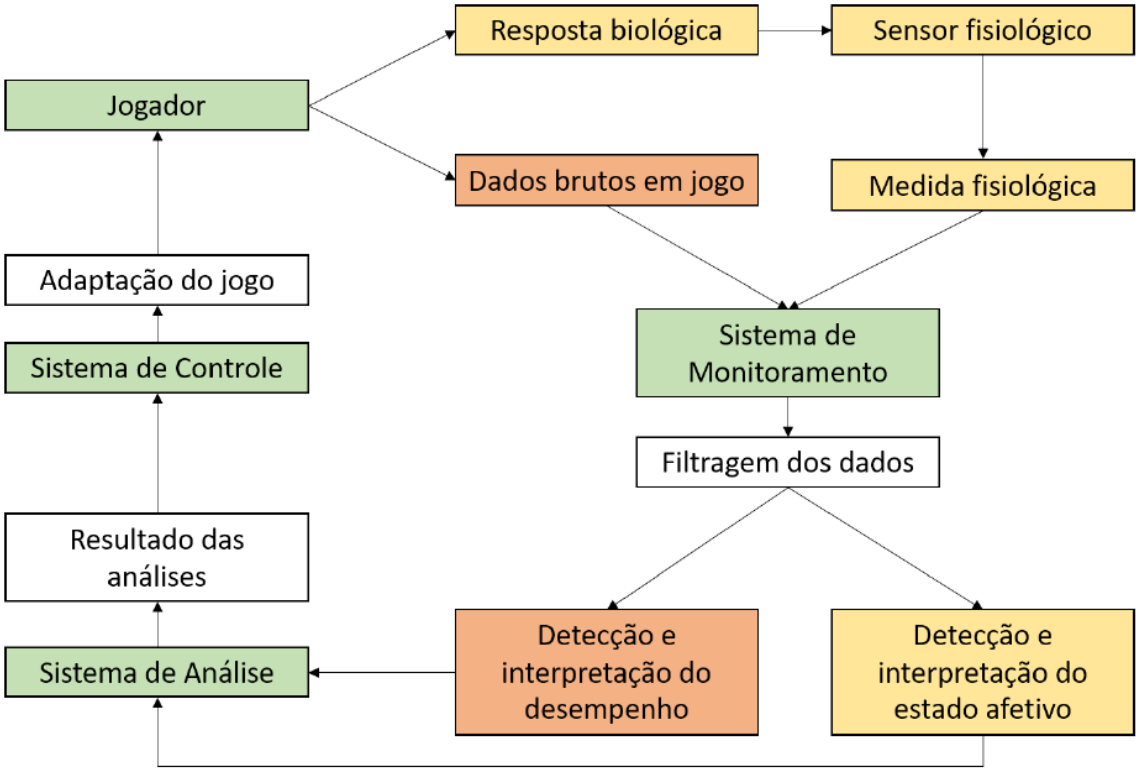

Figura 3.10: Diagrama do Ajuste Dinâmico de Dificuldade híbrido. Os sistemas são representados em verde; as etapas relacionadas à performance em vermelho; e à afetividade, em amarelo [1].

- A décima primeira varia entre 0,16 e 1; e
- A última varia entre 0,25 e 1.

$$maxLevelDif = \frac{level}{numLevels} \tag{3.16}$$

$$\begin{aligned} dif = (dif > maxLevelDif + 0,75) \ ? \ (maxLevelDif + 0,75) : \\ ((dif < maxLevelDif - 0,75) \ ? \ (maxLevelDif - 0,75) : dif) \end{aligned} \tag{3.17}$$

Os ajustes do ADD podem ser feitos em 3 momentos: (1) na criação de segmentos e fases, pois utiliza-se a dificuldade global atual para modificar os componentes; (2) na morte da personagem, pois altera a dificuldade global e aplica as alterações no segmento atual e nos próximos; e (3) em tempo real, toda vez que a dificuldade global for alterada. O modo como os componentes são alterados é o mesmo para todos os casos, sendo seu momento de aplicação a única diferença. Ademais, as modificações não são excludentes.

Por exemplo, é possível que a fase se adapte após a morte da personagem e ao criar novos segmentos e fases.

3.1.6 Ajuste dos componentes pela dificuldade do trabalho de Rosa

Ao inicializar a fase, é feito um mapeamento dos componentes, contendo:

- Dificuldade total;
- Dificuldade total por componentes;
- Sub-mapas de cada segmento; e
- Dificuldade de cada segmento;

Com isso, é possível observar que as dificuldades referentes à fase são definidas pelo somatório do segmento. Ademais, a dificuldade utilizada no ADD é a dos segmentos, pois modifica os seus componentes que ainda estão sendo criados ou altera componentes já existentes.

Cada componente, organizado por tipo, possui:

- Dificuldade associada;
- Direção para cálculo de dificuldade, no caso das plataformas padrão;
- Origem e destino do pulo; e
- Translação executada, no caso das plataformas móveis.

A dificuldade global dif é comparada com a dificuldade do segmento pela quantidade de componentes $segDif$. Essa diferença resulta na dificuldade que deve ser alterada $changeDif$, conforme a Equação 3.18. Varia entre -1 e 1 referente à dificuldade atual, aumentando se for um valor positivo e diminuindo se for um valor negativo.

$$changeDif = dif - segDif \tag{3.18}$$

Na ocasião do valor absoluto da dificuldade a ser alterada seja maior que o limiar de um dos eixos x e y, definidos como $limiar_x$ e $limiar_y$, é averiguado quais componentes específicos devem ser modificados para impedir pulos impossíveis, conforme descrito nas Equações 3.19 e 3.20.

$$limiar_x = \frac{1}{x_r} \tag{3.19}$$

$$limiar_y = \frac{1}{y_v} \tag{3.20}$$

Para não normalizar a dificuldade de todos os componentes de um segmento e manter sua variabilidade, é aplicado um padrão exponencial em cada elemento, tendo:

- $elemDif$ como a dificuldade atual do componente, averiguada antes de qualquer mudança;
- $auxElemDif$ como a dificuldade temporária usada para auxiliar no cálculo; e
- $changeElemDif$ como a dificuldade a ser alterada no componente.

Com isso, todas as fases são variáveis, mas mantém um padrão de design e evita alterações desnecessárias ou exarcebadas, acordante com as Equações 3.21, 3.22 e 3.23.

$$auxElemDif = dif - elemDif \tag{3.21}$$

$$changeElemDif = changeDif^{(1+changeDif-auxElemDif)} \ \forall \ changeDif > 0 \tag{3.22}$$

$$changeElemDif = changeDif^{(1-changeDif+auxElemDif)} \ \forall \ changeDif < 0 \tag{3.23}$$

É importante ressaltar que os cálculos em relação à parábola são realizados no ínicio da fase e só são refeitos caso a velocidade, a aceleração e/ou a gravidade da personagem sejam alteradas, opção fora do escopo tanto do trabalho de Rosa quanto de nossa proposta.

Se o valor absoluto da dificuldade a ser modificada de qualquer componente for maior que o limiar de um dos eixos x e y, o mesmo é alterado em uma quantidade específica de ladrilhos, definidos como qtd_{tiles_x} e qtd_{tiles_y}, respectivamente. A lógica apresentada é demonstrada nas Equações 3.24 e 3.25.

$$qtd_{tiles_x} = \lfloor \frac{changeElemDif}{limiar_x} \rfloor \tag{3.24}$$

$$qtd_{tiles_y} = \lfloor \frac{changeElemDif}{limiar_y} \rfloor \tag{3.25}$$

Modificações no eixo x alteram o tamanho dos abismos, buracos e plataformas. No caso de abismos e buracos, aumenta a quantidade de ladrilhos do destino do pulo se a

dificuldade diminuiu e diminui quando a dificuldade aumenta. Se não houver como alterar o destino, a origem do pulo é afetada.

Modificações no eixo y alteram a posição ou tamanho das plataformas. No caso de plataformas em forma de coluna, são modificadas adicionando ou retirando ladrilhos. Se não puder modificar a quantidade, a posição da plataforma é alterada. No caso das plataformas Inferior, são adicionadas ou movida para cima para aumentar a dificuldade e retiradas ou movidas para baixo para diminuir a dificuldade. As plataformas Superior operam de forma inversa.

Para evitar problemas na geração do mapa e para que o jogo sempre possa ser finalizado, restrições foram estabelecidas com o intuito de manter os padrões de design. Se já possui um ladrilho na posição de adição um novo ou não exista na posição de remoção, nada acontece. Ademais, se uma plataforma possui um conjunto de ladrilhos abaixo ou numa região próxima, ladrilhos com o mesmo padrão serão adicionados para manter a consistência visual do jogo.

Para auxiliar no processo citado e na geração de fases, foi realizado um mapeamento das posições ocupadas e os componentes que as ocupam. Além disso, verifica-se qual é a dificuldade resultante antes de alterar um componente. Portanto, o ajuste só é efetivado quando a dificuldade indica um componente que cumpre o padrão estabelecido de alteração.

As dificuldades referentes aos componentes, o tipo do segmento e da fase são atualizados ao finalizar o procedimento. A Figura 3.11 ilustra a aplicação do ADD, enfatizando a diferença entre dificuldades.

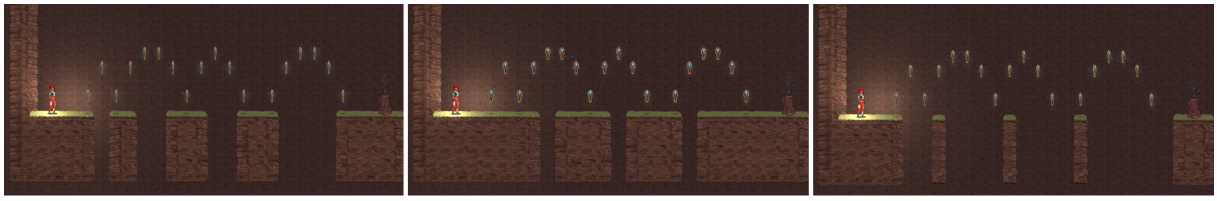

Figura 3.11: Captura de tela do mesmo segmento com diferentes níveis de dificuldade. (a) Sem aplicação de ADD; (b) ADD de performance aplicado com o valor de dificuldade global 0,1; e (c) com o valor em 0,9 [1].

3.2 Proposta

O objetivo da atual pesquisa é desenvolver e analisar se a adição da variável de altura do pulo da personagem afetaria o Ajuste Dinâmico de Dificuldade para jogos de plataforma

da monografia de graduação de Rosa [1] quanto a sua eficácia em manter o jogador em um estado de fluxo. Mais especificamente, as alterações citadas seriam realizadas somente no algoritmo do ADD de performance do ADD híbrido devido à situação de pandemia de COVID-19 no momento em que essa pesquisa foi desenvolvida. Esse contexto dificulta o acompanhamento e a organização de testes presenciais para captar os dados físicos necessários para o ADD afetivo.

Para isso, além de uma revisão extensa do trabalho utilizado como base, um vasto processo de pesquisa verificando estudos relacionados à área de ADD foi realizado. Em torno de 100 artigos relacionados foram verificados e mais de 50 possuiam como seu principal foco a aplicação do ADD. As bases de dados utilizadas para encontrar os artigos citados foram: Google Scholar, IEEE Xplore Digital Library, Association for Computing Machinery (ACM) e Research Gate.

Demonstrar o trabalho de Rosa de maneira extensiva foi de extrema importância, pois o atual estudo utilizará o mesmo algoritmo do ADD de performance, que focava na alteração do terreno. Portanto, as fórmulas e conceitos apresentados continuam válidos.

Em primeira instância, o foco será verificar se o novo ADD proposto, o ADD de pulo, está devidamente implementado, sem interferir com os cálculos do ADD de performance do trabalho de Rosa, o qual será referenciado como ADD de plataforma. Para manter a consistência entre os trabalhos e ser uma comparação válida, a coleta de dados e momento de aplicação do ADD serão iguais ao trabalho de Rosa, ou seja, os dados são coletados e analisados em tempo real, mas a aplicação do ajuste é feita somente quando o jogador morre ou completa um segmento de fase ao alcançar um *checkpoint* ou portal. Além disso, como os testes desse estudo precisam ser online, é necessário uma primeira bateria de testes para ajustes e correção de bugs, visto que o projeto original não foi adaptado para rodar em várias máquinas distintas. Com isso executado, o objetivo é comparar 4 casos diferentes e determinar se a sua eficiência em manter o jogador em estado de fluxo será afetada ao alterar a característica intrínseca da personagem de amplitude do pulo com base na performance do usuário. Os casos são:

1. Sem aplicação de ADD;
2. Aplicação de ADD de performance de plataformas (implementado por Rosa);
3. Aplicação de ADD de performance de pulo; e
4. Aplicação de ADD de performance combinado de plataforma e pulo.

Adaptações

A primeira adaptação feita no projeto foi alterar o fluxo dos questionários. A apresentação de perguntas no meio da sessão para o usuário como feito no projeto de Rosa [1]

possibilitava a quebra do estado de fluxo, um resultado indesejado para este estudo. Com o intuito de aumentar a quantidade de jogadores em estado de fluxo, foi decidido não interromper o ato de jogar as fases durante a sessão. Ao iniciar a sessão, o foco do usuário não se encontra nos controles nem nos desafios, pois ainda não começou a jogar pelas fases, sendo um excelente momento para adicionar um questionário inicial. Ao finalizar a última fase, o jogador pode ser apresentado a um questionário, visto que os desafios encerram e a análise de seu desempenho já foi concluída, portanto não há como essa pausa na ação causar efeitos colaterais em sua performance. Portanto, os questionários foram posicionados no início da sessão de jogo antes de carregar a primeira fase e ao finalizar o jogo.

A segunda adaptação feita no projeto foi a implementação do Ajuste Dinâmico de Dificuldade de pulo, que influencia a amplitude do salto da personagem conforme o desempenho do usuário. Como no ADD de plataforma já existiam contadores de morte e de pulos de sucesso e fracassados, as mesmas variáveis foram utilizadas como parâmetro para as modificações. Portanto, o cálculo da performance para todos os casos com aplicação de ADD utiliza a quantidade de mortes e de pulos. Adicionalmente, cada sessão passou a possuir apenas uma aplicação de ADD durante todas os 12 níveis, sem nenhuma interrupção ao longo da mesma.

A personagem inicia a sessão de jogo com o valor padrão de amplitude do pulo, portanto o valor global de dificuldade começa com a média, valor 0,5. Ao passar por um *checkpoint* ou ao morrer, uma nova varíavel $difFactor$, que auxilia na normalização da dificuldade global para o ADD do pulo, recebe um valor fixo. No caso de morte, 0,01. No caso de ajuste por segmento (transitar por um *checkpoint*), 0,005. Para definir a dificuldade, é preciso estabelecer a proporção de pulos de sucesso e falhos. Definindo p como a proporção de pulos de sucesso, $jumpSucess$ como o valor de pulos de sucesso e $jumpFail$ como o valor de pulos fracassados, a Equação 3.26 é obtida:

$$p = (jumpFail <= 0) \ ? \ jumpSucess : \frac{jumpSucess}{jumpFail} \tag{3.26}$$

Se a proporção de pulos de sucesso for maior que o limiar (de valor 1 para este estudo) referente aos pulos fracassados, a dificuldade global é incrementada com a multiplicação do número de pulos de sucesso e o fator de dificuldade $difFactor$. Se for menor ou igual, a dificuldade global é decrementada com o fator de dificuldade. Caso a dificuldade global esteja em seus extremos, ou seja, com 0 ou 1, nada acontece.

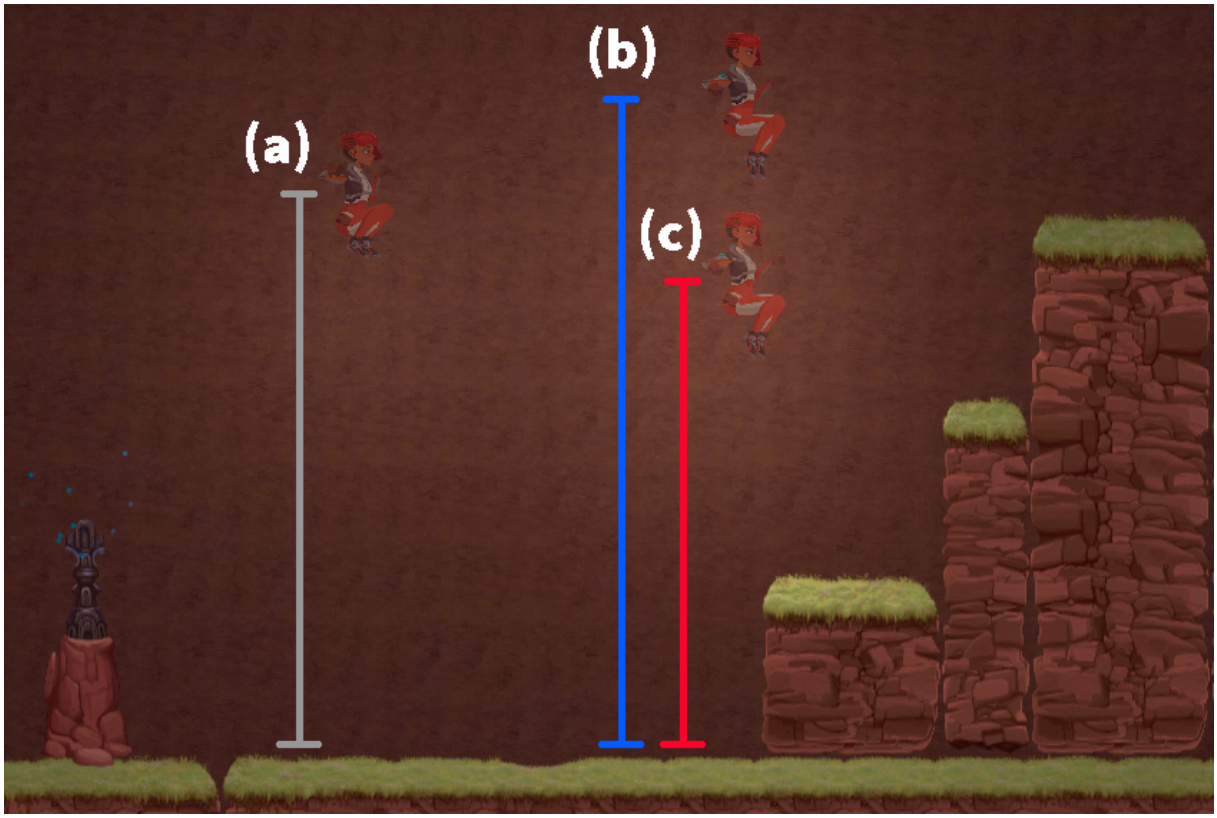

Figura 3.12: Captura de tela do mesmo segmento com as diferentes amplitudes de pulo. (a) Amplitude máxima sem ADD aplicado; (b) Com ADD de pulo aplicado com o valor de dificuldade global 0, ou seja, mais fácil; e (c) Com o valor em 1, ou seja, mais difícil.

Coleta de Dados

Semelhante ao trabalho de Rosa [1], ao encerrar uma sessão, os dados de desempenho, assim como as respostas de ambos os questionários, são armazenados localmente. Foi acrescentado uma variável para indicar qual dentre os 4 casos possíveis foi o caso jogado. Além disso, os dados necessários do teste foram comprimidos em um único arquivo, facilitando para os participantes da pesquisa enviarem os dados corretamente por meio de um formulário.

Capítulo 4

Testes e resultados

Neste capítulo serão descritos os testes executados com voluntários e analisados os principais resultados obtidos a partir desses.

4.1 Testes

O foco deste estudo é avaliar a eficiência do modelo de ADD por desempenho utilizado por Rosa [1] quando aplicado em elementos distintos de jogo, assim realizando uma comparação entre os casos quanto a qual seria mais eficaz em manter o jogador no fluxo e também em permitir uma melhor percepção de dificuldade adequada. Portanto, é feita uma correlação entre a dificuldade mensurada pelo algoritmo e a experienciada pelos jogadores.

Foram realizadas duas baterias de teste com voluntários. Participantes da primeira bateria não poderiam participar da segunda bateria, assim evitando qualquer viés que pudesse surgir da experiência prévia do indivíduo com o jogo. Ambos apresentavam um total de 12 níveis, cada um contendo de três a quatro segmentos. A dificuldade do jogo foi crescente e analisou, em cada nível, diversos elementos de jogo e suas conjunções.

Com os testes piloto, pretendia-se garantir que o funcionamento do jogo estivesse correto, assim como analisar e garantir que a organização utilizada nesta bateria permitisse uma qualidade e integridade dos dados resultantes. Ao total foram 12 participantes na primeira bateria, com 3 jogadores para cada caso analisado. Através disso, permitiu-se averiguar se o que foi determinado para a primeira bateria poderia ser utilizado também na segunda, a qual seria composta pelos testes definitivos.

O modelo de testagem utilizado nos testes piloto e replicado em maior escala nos testes definitivos baseou-se em sortear ao jogador, durante o início da partida, um elemento que seria ajustado pelo ADD. As opções eram: o ADD aplicado a plataformas, aplicado ao pulo da personagem, aplicado a ambos e, por fim, sem aplicação de ADD. Com a transição de

segmentos ou com a morte da personagem, o algoritmo de ajuste era aplicado e, portanto, a dificuldade alterada.

A escolha dos elementos e obstáculos em cada conjunto foi feita para evitar que aspectos particulares enviesassem a conclusão, assim eliminando a existência de chefes ou de ambientes exploratórios. Os quatro casos foram comparados por meio dos dados resultantes do algoritmo e de questionários aplicados ao voluntário. A partir disso, formulou-se a seguinte hipótese:

- O Ajuste Dinâmico de Dificuldade combinado com a adaptação das plataformas e da amplitude do pulo da personagem mantém o jogador no estado de fluxo e provê uma dificuldade adequada, com resultados superiores ao sistema que adapta somente as plataformas.

4.1.1 Descrição dos níveis e execução dos testes

Em ambas as baterias, utilizou-se os mesmos níveis, sendo que estes foram retirados do trabalho de Rosa [1]. Divididos em níveis fáceis, médios e difíceis, cada nível continha os elementos listados a seguir, sendo estes incrementais entre as dificuldades dos níveis:

- **Níveis fáceis:** chave, buraco, *checkpoint*, escada, objeto quebrável, plataforma básica, plataforma capaz de atravessar e portal de saída;
- **Níveis médios:** ácido, objeto que se empurra, inimigo com ataque corpo a corpo, buracos maiores e plataforma móvel de movimento vertical; e
- **Níveis difíceis:** espinhos, buracos maiores e com menor espaço no destino, plataforma móvel de movimento horizontal e diagonal, inimigo com ataque à distância e plataformas com espaço mínimo pra locomoção.

Os testes foram validados pela aplicação de questionários em certos momentos da sessão, esses presentes em sua totalidade no Apêndice A. As questões aplicadas antes do jogo iniciar se referiam a dados sociodemográficos. Ao final, analisou-se a dificuldade e a imersão percebidas e se o jogador se manteve em estado de fluxo.

Tanto os testes piloto quanto os testes definitivos foram realizados de maneira remota, em decorrência da pandemia de COVID-19. Por meio de um formulário *Google Forms*, o voluntário era apresentado os termos da pesquisa e inseria alguns dados como e-mail e nome (para garantir que ele só realizasse o teste uma única vez). Em sequência, era permitido baixar o jogo através de um *link* fornecido. Os dados de desempenho e dos questionários foram coletados e salvos em cada computador localmente, os quais eram enviados pelo usuário para os pesquisadores através do formulário. A estrutura do formulário mais detalhada é apresentada abaixo:

1. Apresentação do Termo de Consentimento Livre e Esclarecido (TCLE);
2. Coleta de dados pessoais (nome, e-mail) visando garantir a unicidade do indivíduo no teste;
3. Permitir ao usuário prosseguir apenas com a concordância com os termos previamente citados;
4. Apresentar ao jogador o *link* para *download*, assim como instruções de como executar o jogo e como enviar os dados resultantes; e
5. Apresentar ao jogador um campo para envio dos dados (os quais eram gerados e compactados pelo próprio jogo).

4.2 Resultados

Na segunda bateria de testes, participaram um total de 155 pessoas. Os participantes possuíam idade variando entre 15 e 65 anos, com uma mediana de 25 anos. 81,69% se identificaram como sendo do gênero masculino, 17,61% como do gênero feminino e 0,7% como de outros gêneros. Avaliou-se o quanto cada jogador se considerava casual ou dedicado em uma escala de 1 a 5, com a maior incidência sendo de jogadores "quase dedicados", ou seja, que se indentificavam na quarta posição dessa escala. Quanto a familiaridade e apreço à jogos de plataforma, estas se mantiveram entre um valor médio e alto. Observou-se também a predileção entre os jogadores pela dificuldade Média (51,41%), seguido pela Difícil (32,39%) e pela Fácil (16,20%). Constatou-se que os participantes jogam com maior frequência no computador pessoal, seguido a distância por celulares, consoles de mesa e, por fim, consoles portáteis.

Após análise da base de dados, um total de 11 testes foram descartados por se encaixarem como *outliers*. O critério para definir se cada teste era ou não descartado foi baseado nos seguintes pontos:

- Descartar participantes que preencheram dados demográficos incorretamente; ou
- Descartar se o jogador tiver desistido de um nível em um intervalo menor que 10 segundos e tendo morrido uma ou menos vezes ao longo deste (revelando que o jogador, potencialmente, não estava empenhado na tentativa); ou
- Descartar se o jogador tiver passado um tempo maior que 1000 segundos tentando completar o nível sem mortes (um possível indicador do jogador manter a sessão de jogo aberta sem estar jogando).

Através dos testes adquiridos, realizou-se a organização e análise dos dados coletados. Dentre os testes restantes, a divisão entre casos de aplicação sorteados foi tal que haviam 37 casos sem aplicação de ADD, 29 casos de aplicação de plataforma, 35 casos de aplicação de pulo e 31 casos de aplicação combinada. Ao fim da partida, era requisitado ao jogador que atribuísse um valor de concordância (estipulado de 1 a 5) sobre as afirmações apresentadas, estas que tinham como objetivo permitir a análise de experiência e imersão. A Figura 4.1 apresenta um comparativo da média dessas respostas em relação a cada caso de aplicação.

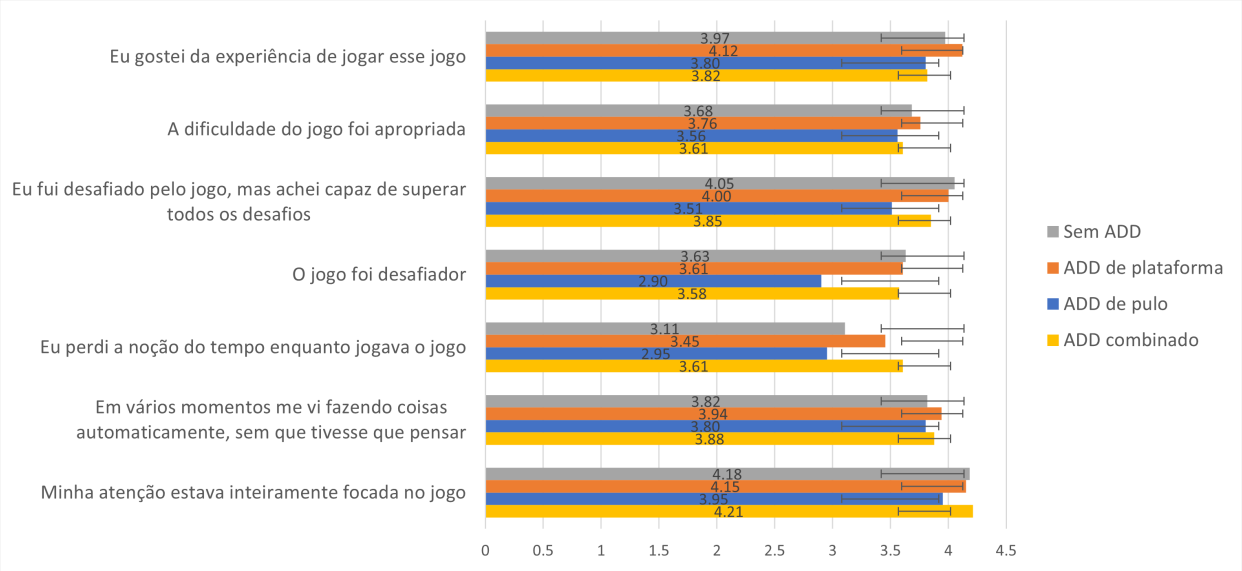

Figura 4.1: Valores médios e desvio padrão das respostas sobre experiência e imersão de todos os participantes na segunda bateria de testes, separados por caso de ADD selecionado.

Quanto a qualidade da experiência percebida pelos usuários ao jogar, é notável que o ADD aplicado somente às plataformas possui um desempenho melhor em comparação ao demais modelos. Percebe-se também que o jogo sem nenhuma aplicação de ADD ainda rende uma experiência melhor do que ao ADD aplicado ao pulo ou composto. Sobre a dificuldade do jogo ser apropriada, percebe-se também que a melhor aplicação seria às plataformas, seguidas por nenhuma aplicação. Novamente, o ADD aplicado à ambos os elementos e o aplicado somente ao pulo possui um desempenho menor.

Em termos de desafio e a percepção do jogador quanto a sua capacidade de superá-lo, a aplicação à plataforma e nenhuma aplicação aparecem à frente, seguidas pela aplicação combinada. O ADD por pulo possui um rendimento menor ao proporcionar um jogo desafiador, porém todos permanecem acima do valor médio da escala. Quanto ao jogo ser desafiador, o desempenho de todos as aplicações foi bem próximo, com exceção da

aplicação somente ao pulo, a qual ficou inclusive levemente abaixo do valor médio da escala.

Voluntários relataram a aplicação combinada como a que melhor resultou em manter o estado de fluxo do jogador e na perda da noção de tempo, assim como a que melhor permitiu um foco total da atenção no jogo. O ADD aplicado apenas a plataforma permitiu ao jogador melhor automação em suas ações, porém apenas por uma margem muito pequena em relação à aplicação combinada. No geral, os valores medidos mostram que os jogadores experienciaram mais o estado de fluxo através do ADD combinado, seguido pelo ADD de plataforma. Os casos onde nenhum ADD foi aplicado vem em seguida, com a aplicação somente ao pulo possuindo uma menor avaliação.

Avaliou-se a percepção dos jogadores quanto à mecânica de pulo no jogo, como pode ser visualizado na Figura 4.2. Ao comparar-se os casos de aplicação, o ADD combinado revelou-se aquele cujo o pulo seria o mais difícil de controlar. Foi seguido pela ausência de ADD, pelo ADD de plataforma e, por fim, pelo ADD de pulo, considerado o mais fácil.

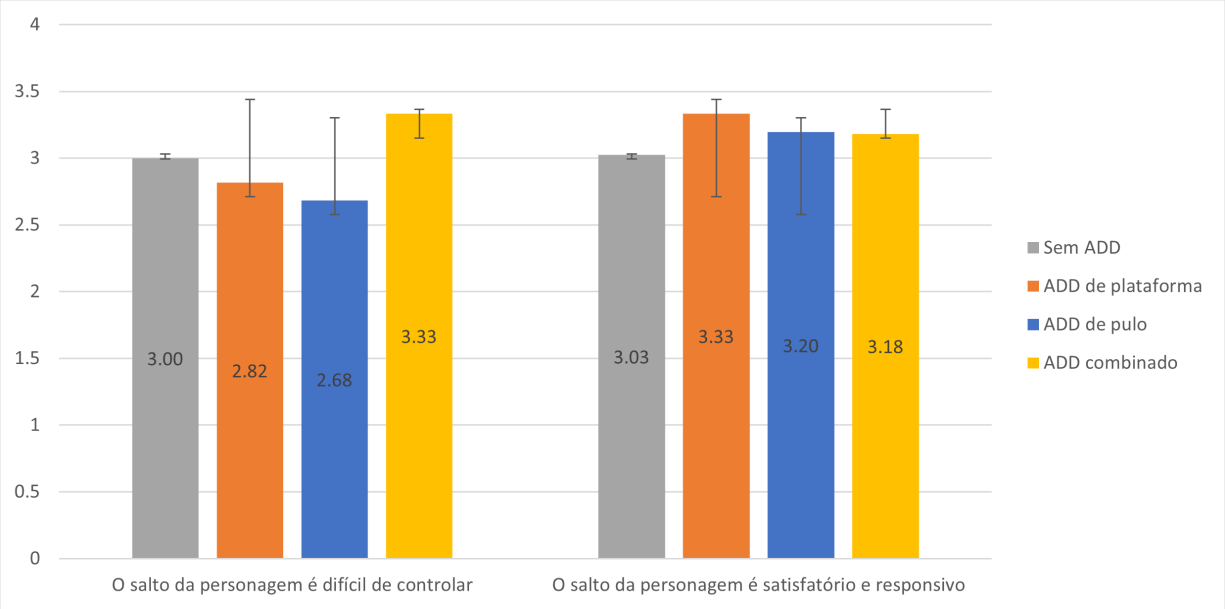

Figura 4.2: Valores médios e desvio padrão das respostas sobre a percepção quanto ao pulo da personagem de todos os participantes na segunda bateria de testes, separados por caso de ADD selecionado.

Em termos de responsividade e satisfação gerada pelo salto, o ADD de plataforma tem um leve desempenho superior, porém possui também um desvio padrão alto, mostrando que há discordância entre os participantes quanto a esta resposta. Por sua vez, o ADD combinado possuiu um desempenho levemente menor, mas com um desvio padrão bem menor, o que indica a consistência de sua qualidade neste aspecto. O ADD de pulo vem em seguida, possuindo também um alto desvio padrão. Por fim, a ausência de ADD possui

Tabela 4.1: Porcentagem de níveis que foram completados por cada usuário categorizados por tipo de ADD escolhido.

Nível	**Sem ADD**	**ADD de plataforma**	**ADD de pulo**	**ADD combinado**
1	100,00%	96,88%	97,56%	100,00%
2	97,44%	98,75%	97,56%	100,00%
3	97,44%	96,88%	97,56%	100,00%
4	100,00%	96,88%	97,56%	96,97%
5	94,87%	90,63%	95,12%	90,91%
6	97,44%	96,88%	92,68%	96,97%
7	97,44%	96,88%	97,56%	96,97%
8	97,44%	96,88%	97,56%	93,94%
9	87,18%	90,63%	85,37%	84,85%
10	92,31%	96,88%	85,37%	87,88%
11	97,44%	96,88%	85,37%	87,88%
12	94,87%	96,88%	85,37%	93,94%

o menor desempenho e a maior disperção analisada a partir de seu desvio padrão. Tais dados revelam detalhes interessantes: o ADD de pulo torna a mecânica mais fácil, porém levemente menos responsiva e satisfatória. Além disso, fica claro como a responsividade por parte da mecânica de pulo possui um desempenho mais relevante, pois, mesmo sendo o caso onde tal elemento se torna mais difícil, é também o caso onde se há os maiores e mais consistentes dados a respeito da satisfabilidade e responsividade.

Analisou-se a porcentagem de conclusão de todas as fases jogáveis para cada caso de aplicação, conforme demonstrado na Tabela 4.1. É possível notar que uma quantidade razoável de jogadores completou os níveis 1 a 8, categorizados como níveis fáceis e médios. Entretanto, no início das fases difíceis, o nível 9, observa-se a menor porcentagem de conclusão de níveis para todos os casos aplicados. Tais dados sugerem que a estrutura da fase 9, ou seja, seu *level design* é frustrante independentemente da aplicação de ADD, afetando o estado de fluxo do jogador. Entretanto, o ADD combinado possui o pior rendimento, contendo o menor valor de conclusão da tabela inteira, com 84,85% de participantes finalizando o nível 9.

Além da menor porcentagem de conclusão de fase, o nível 9 também possui dois dos elementos apontados como mais desafiadores pelos usuários no questionário após a finalização da sessão de jogo: espinhos e inimigos com ataque a distância. Separados pelas dificuldades preferidas dos usuários, em ambas as Figuras 4.3 e 4.4, eles obtiveram uma pontuação igual ou maior que 3 para todos os casos, indicando que, independentemente da preferência do jogador, foram elementos desafiadores que influenciaram o estado de fluxo do jogo.

Quanto ao tempo de permanência dos usuários em cada fase, foram organizados e

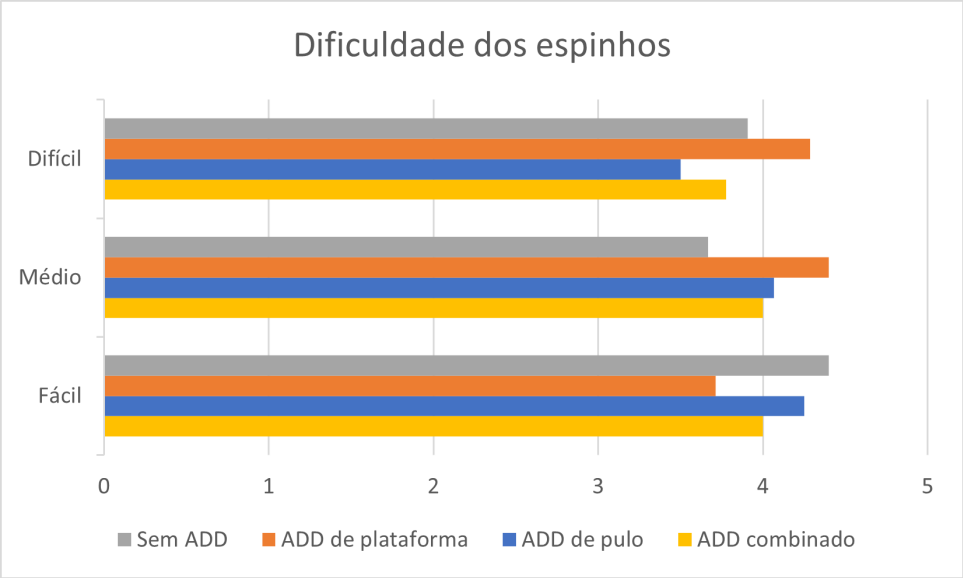

Figura 4.3: Média das respostas determinando o quanto os espinhos foram desafiadores em sua experiência em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

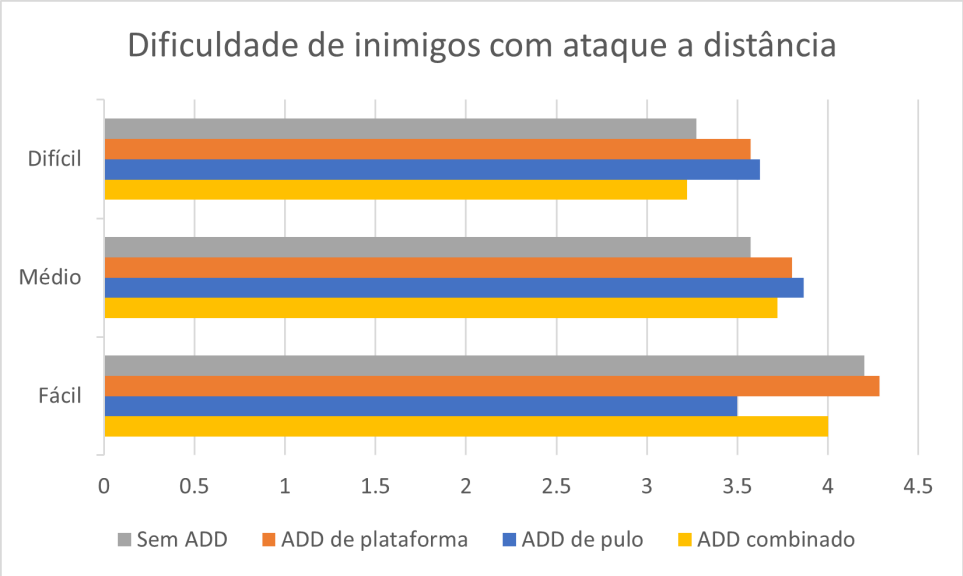

Figura 4.4: Média das respostas determinando o quanto os inimigos com ataque a distância foram desafiadores em sua experiência em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

separados por tempo mínimo, tempo médio, tempo máximo e mediana de todos os casos aplicados. Removendo os dados inválidos, é obtida a Tabela 4.2. Vale observar que os valores máximos demonstrados na tabela ainda são válidos, apesar de passarem de 1000 segundos, devido a quantidade proporcional de variáveis de contagem de pulo e

Tabela 4.2: Tempo mínimo, médio, máximo e mediana relacionados a permanência em cada nível da segunda bateria.

Nível	**Tempo Mínimo**	**Tempo Médio**	**Tempo Máximo**	**Mediana**
1	15,54	60,18	333,11	50,65
2	17,05	32,92	151,98	31,24
3	20,76	43,38	199,16	36,16
4	16,10	42,35	182,16	37,43
5	19,49	80,16	3589,29	47,65
6	17,12	47,85	457,05	45,29
7	29,58	67,13	366,06	61,19
8	17,53	59,21	228,36	61,45
9	39,04	189,11	2904,77	185,77
10	34,88	120,01	3806,28	103,46
11	26,28	73,16	3992,26	56,29
12	36,08	80,43	233,05	89,51

movimentação dessas tentativas.

As fases foram feitas para durar entre 30 segundos e 2 minutos, variando conforme a dificuldade da fase em questão. Analisando os dados da média, é possível averiguar que a meta foi atingida, exceto pelo nível 9. Além disso, o mesmo possui a maior mediana apresentada na tabela, provando mais uma vez sua irregularidade, ocupando mais tempo de diversos jogadores do que qualquer outro nível.

Com os dados coletados e organizados do Apêndice A, foi possível separar os grupos de jogadores pela dificuldade preferida e os casos que foram jogados. Averiguando as diversas respostas, observa-se que o ADD de pulo foi mais efetivo para usuários que preferem jogos fáceis. Essa afirmação se baseia nas respostas da dificuldade do pulo e em sua responsividade, na quais o ADD de pulo para jogadores casuais foi a melhor opção entre os casos, tendo a menor dificuldade no pulo e melhor responsividade, conforme ilustrado nas Figuras 4.5 e 4.6. Ademais, também foi o ADD que demonstrou ter o desafio mais justo, dificuldade mais apropriada e com maior satisfação ao jogar pelos participantes, conforme demonstrado nas Figuras 4.7, 4.8 e 4.9.

Ao registrar as dificuldades que cada participante teve em cada fase para todos os modelos, foi possível realizar a média de cada caso, obtendo a Figura 4.10. Observa-se que os gráficos possuem linhas de tendência parecidas, ou seja, picos e valores proporcionais, demonstrando um padrão semelhante. Portanto, se infere que a aplicação de qualquer ADD presente nesse estudo aumenta a dificuldade da maioria das fases, visto que o caso sem aplicação de ADD, com exceção da fase 2, possui a menor média registrada de dificuldade para todas as fases de todos os modelos.

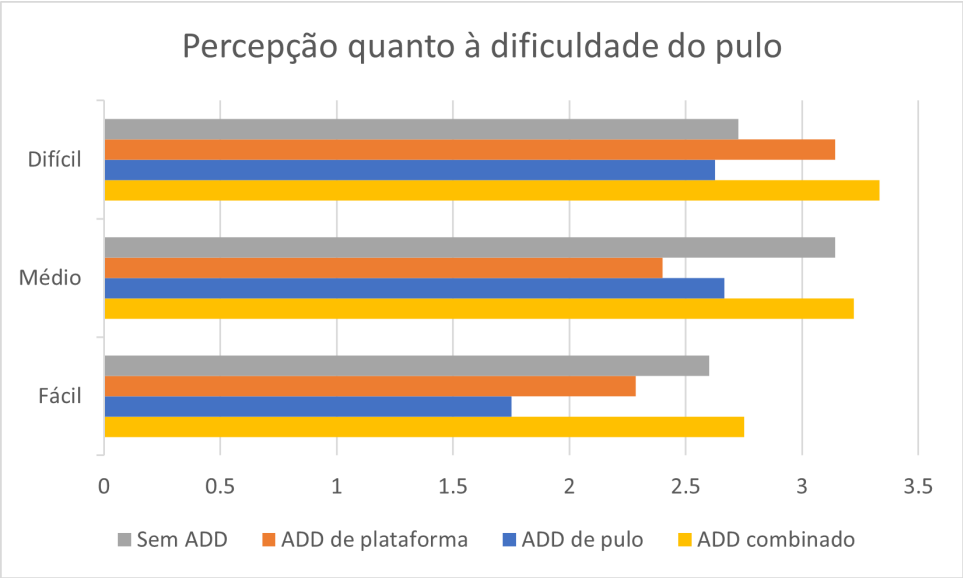

Figura 4.5: Média das respostas determinando a dificuldade que o jogador teve em controlar o pulo da personagem em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

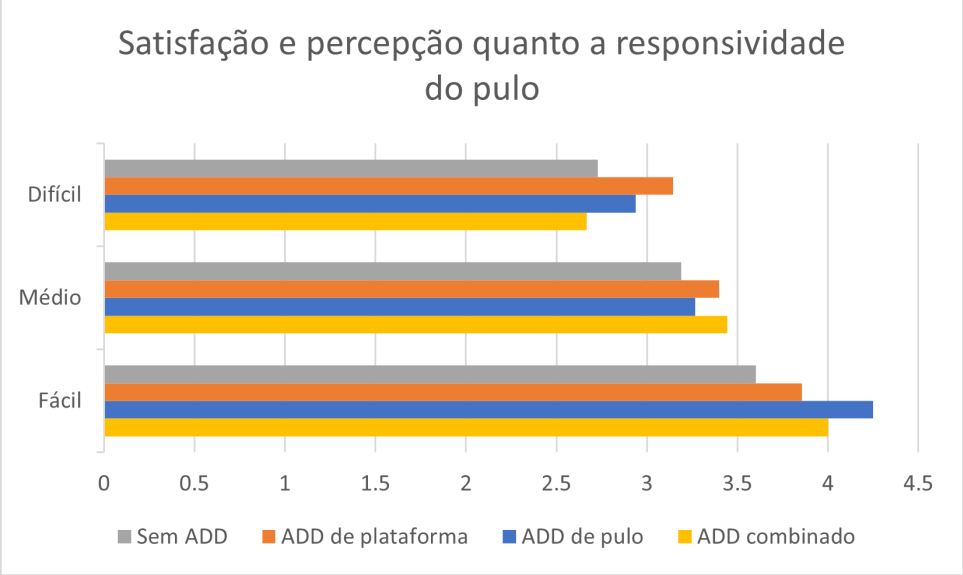

Figura 4.6: Média das respostas determinando a satisfação e percepção da responsividade do pulo para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

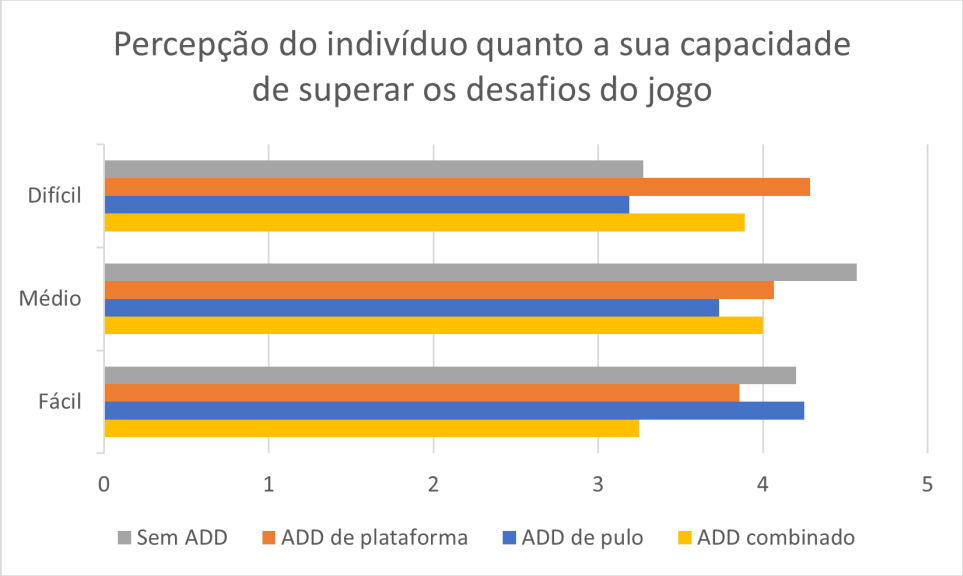

Figura 4.7: Média das respostas determinando se o desafio foi justo para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

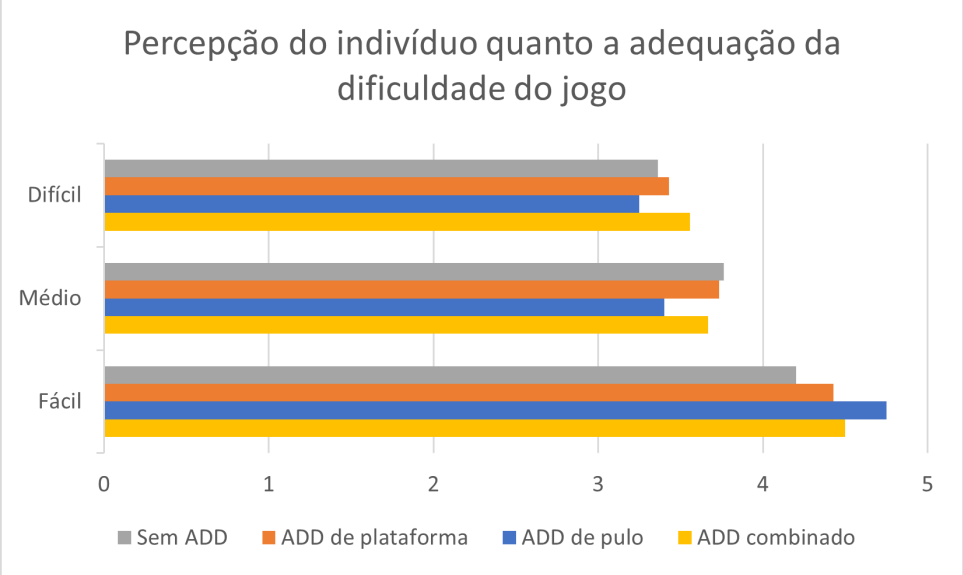

Figura 4.8: Média das respostas determinando se a dificuldade foi apropriada para o jogador em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

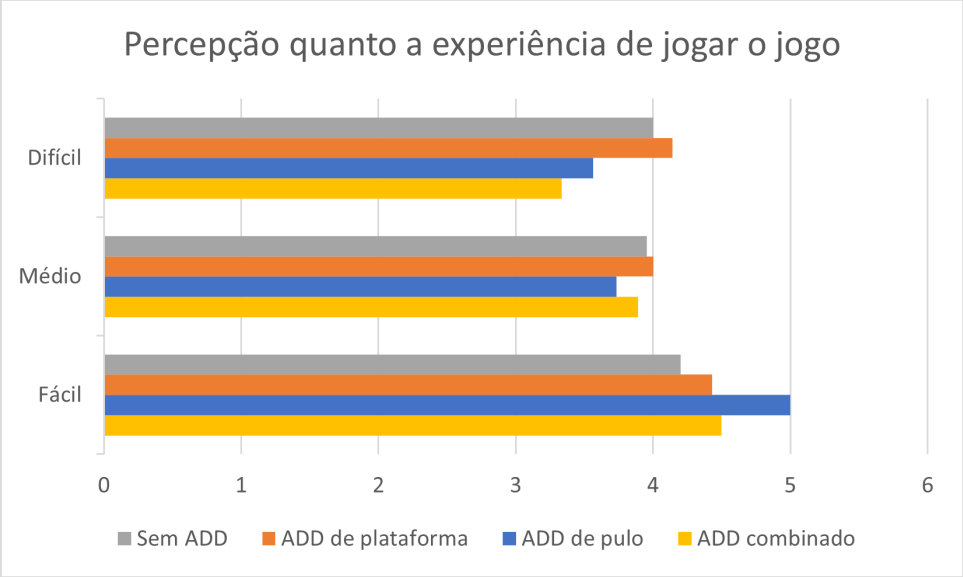

Figura 4.9: Média das respostas determinando o quanto o jogador gostou da sua experiência com o jogo em uma escala de 1 a 5, relacionadas com a dificuldade preferida dos usuários.

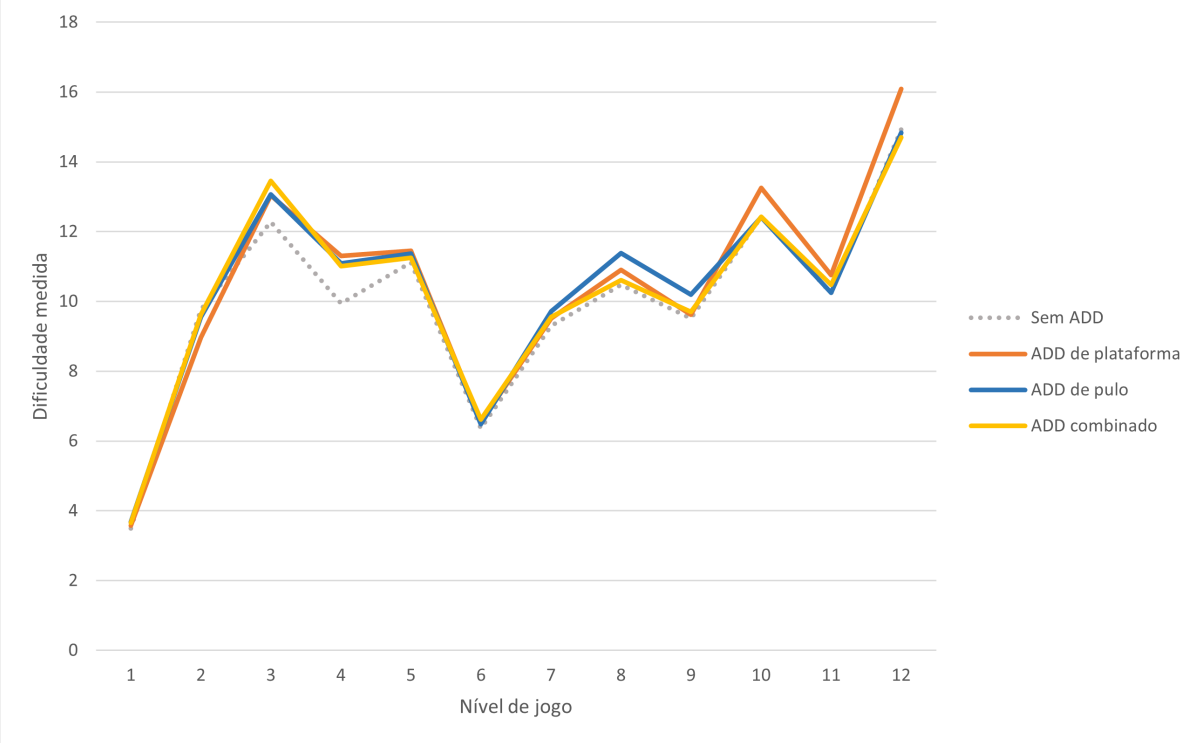

Figura 4.10: Média das dificuldades medidas durante as partidas de cada voluntário por nível, separado por modelo de ADD.

Capítulo 5

Considerações Finais

Este capítulo sintetiza a presente pesquisa e as principais conclusões da análise realizada, assim como propõe novos desafios para estudos futuros na área.

5.1 Conclusões

A área de Ajuste Dinâmico de Dificuldade (ADD) possui uma grande abrangência de estudos, com ajustes conforme o desempenho do jogador, o estado afetivo do jogador ou ambos em conjunto. Especificamente para o gênero plataforma, o ajuste híbrido foi analisado somente pelo trabalho de Rosa [1]. Entretanto, nenhuma pesquisa na literatura efetuava uma alteração em alguma característica intrínsica do avatar controlado pelo jogador de acordo com a avaliação de seu desempenho. Por característica intrínsica, denomina-se qualquer ação padrão do avatar, como se movimentar, agachar ou pular.

Desse modo, esta pesquisa teve como foco desenvolver um modelo de ADD combinado, o qual adapta tanto a amplitude do pulo do avatar controlado pelo jogador quanto às plataformas presentes em um jogo de mesmo gênero. Mais especificamente, o objetivo foi ajustar a dificuldade com base nos dados relativos ao desempenho de cada jogador e mantê-los em estado de fluxo. Algumas modificações foram feitas no jogo utilizado no trabalho de Rosa, que por sua vez é uma versão alterada do jogo de uso livre *The Explorer: 2D*, desenvolvido pela empresa *Unity Technologies*. Foram utilizadas as alterações de geração de fases, exportação de dados internos do jogo e ajuste de plataformas. As novas modificações consistem em ajuste da amplitude de pulo da personagem, adição de dados relacionados ao ADD de pulo aos dados exportados e a escolha aleatória de um dos quatro casos estudados para jogar:

1. Sem ajuste;
2. Ajuste nas plataformas da fase, com base no desempenho do jogador;

3. Ajuste na amplitude do pulo do avatar, com base no desempenho do jogador; e
4. Combinação dos modelos de ajuste nas plataformas da fase e na amplitude do pulo do avatar.

Para todos os casos com aplicação de ADD, a performance do usuário foi medida utilizando os dados de suas mortes e sua execução de pulos. Embora os dados fossem coletados e analisados em tempo real, as alterações somente eram aplicadas após a morte do jogador ou ao completar um segmento, ou seja, ao encostar em um *checkpoint* ou portal.

Para aplicação do ajuste, foi utilizado o mesmo método de estimativa de dificuldade do trabalho de Rosa, que define a dificuldade de uma fase como o somatório da dificuldade de diversos componentes de plataforma, adquiridos pela probabilidade de sucesso nos pulos executados.

Conduziu-se duas baterias de testes com voluntários, sendo que participantes da primeira bateria não participaram na segunda bateria. Ambos os testes foram realizados de forma remota, com um texto de apresentação e explicação de como realizar o teste. A primeira bateria serviu para averiguar problemas no projeto e na apresentação das instruções, refinando para a segunda bateria, que teve um alcance muito maior. Jogadores de diversos níveis de habilidade com jogos se voluntariaram para a pesquisa. Dados sociodemográficos foram coletados ao iniciar a sessão de jogo e um questionário foi aplicado ao término da mesma.

Conforme os resultados apresentados e avaliados na Seção 4.2, observa-se diversas informações. Ao analisar as respostas sobre a experiência e imersão de todos os tipos de jogadores, nossa hipótese não é confirmada, pois apresenta conclusões multifacetadas. Quanto ao estado de fluxo, o ADD combinado mostrou-se mais eficiente, seguido pelo ADD de plataforma. O ADD combinado foi o melhor em fazer os jogadores perderem a noção de tempo, assim como o com mais jogadores com foco total da atenção no jogo. O ADD de plataforma foi mais eficiente para a automação das ações do jogador, mas por uma margem muito pequena em relação ao ADD combinado. Para jogadores que preferem jogar na dificuldade fácil, constatou-se que o ADD de pulo foi o mais eficiente e satisfatório, possuindo os melhores resultados médios.

Verificando as tabelas e gráficos sobre porcentagem de conclusão de fases, tempos de permanência em cada fase e componentes mais desafiadores nas fases, observa-se que a fase 9 afetou o estado de fluxo de todos os quatro casos estudados. Esse nível apresenta a menor porcentagem de jogadores que concluíram em todos os modelos e a maior mediana de tempo para conclusão de todas as fases. Observa-se que os componentes considerados mais desafiadores, espinhos e inimigos com ataque a distância, são introduzidos nesse

mesmo nível. Portanto, esse fase deve ser descartada ou revisada para não afetar os dados da forma que ocorreu.

Averiguando as respostas sobre a percepção quanto ao pulo da personagem de todos os participantes, nota-se que o ADD de pulo torna essa mecânica mais fácil, entretanto um pouco menos responsiva e satisfatória. Ademais, observa-se como a responsividade da mecânica de pulo possui uma performance mais relevante, porque mesmo sendo o caso onde ele se torna mais difícil, é também o caso onde os dados são os maiores e mais consistentes referentes à satisfação e responsividade.

5.2 Trabalhos futuros

Para trabalhos futuros, possíveis objetos de estudo são: acrescentar análise do estado afetivo aos cálculos do ADD de pulo, verificações aprofundadas do cálculo de performance e expandir a aplicação do Ajuste Dinâmico de Dificuldade para afetar outros componentes e integrar com os sistemas já desenvolvidos. Componentes que ainda não foram explorados são:

- Inimigos;
- Obstáculos (espinhos, caixas de madeira, pilares, pedras);
- Outras características da personagem:
    - Velocidade (movimentação, sacada de espada, tiro);
    - Aceleração de movimentação; e
    - Alcance (sacada de espada, tiro).
- Posição da câmera; e
- Sistema de vidas.

Aumentar o intervalo de dificuldade permitido pelo ADD seria outro possível elemento de trabalho futuro, pois assim seria possível haver a coleta de novos dados que possuam maior clareza em suas variáveis, o que permitiria averiguar melhor o funcionamento do ADD. Também seria de grande valor implementar um ADD cujo o funcionamento dependa de uma preferência de dificuldade informada pelo usuário previamente (tal que se o jogador prefira a dificuldade fácil, média ou difícil, o algoritmo leve essa informação em conta para dar uma dificuldade como resultado).

Outro aspecto a ser analisado é o momento de aplicação do ajuste, visto que pode influenciar no estado de fluxo do jogador. Alguns desses momentos são ao gerar segmentos

e fases, ao morrer e em tempo real, aplicando a modificação durante um determinado intervalo de tempo. Seria interessante também realizar a execução de testes estatísticos mais precisos em cima dos dados coletados e analisados neste trabalho, assim entendendo melhor a significância das variáveis envolvidas.

Por fim, um objeto de estudo possível é analisar a correlação entre o ADD e *level design*. Desse modo, permitiria a verificação de se uma má estrutura de nível é suficiente para afetar completamente o estado de fluxo de seus jogadores ou se um ADD poderia reverter essa situação. Tal *level design* poderia também afetar a medida de variáveis relacionadas ao ADD, dificultando sua análise e, portanto, tornando ainda mais interessante a realização de uma averiguação quanto a esse aspecto.

Referências

[1] Rosa, Marcos Paulo Cayres: *Ajuste dinâmico de dificuldade híbrido em um jogo do gênero plataforma.* Dissertação de Graduação, Universidade de Brasília, 2019. v, vi, ix, x, xi, 2, 7, 13, 14, 22, 23, 24, 25, 27, 28, 29, 30, 32, 33, 34, 36, 41, 44, 47, 48, 50, 51, 52, 62

[2] Silva Valentim Mourato, Fausto José da e Manuel Próspero dos Santos: *Measuring difficulty in platform videogames.* 4a Conferência Nacional em Interação Pessoa-Máquina, 2010. ix, 9, 24

[3] Ben Cowley, Darryl Charles, Black Michaela e Hickey Ray: *Toward an understanding of flow in video games.* Computers in Entertainment, 2008. ix, 11, 12, 13

[4] Araujo, Bruno Baère Pederassi Lomba de: *Um estudo sobre a adaptatividade dinâmica de dificuldade em jogos.* Dissertação de mestrado, Pontíficia Universidade Católica do Rio de Janeiro, 2012. ix, 13, 14, 16

[5] Silva Valentim Mourato, Fausto José da: *Enhancing automatic level generation for platform videogames.* Tese de doutorado, Universidade Nova de Lisboa, 2015. ix, 7, 8, 9, 15, 34, 39

[6] *Videogames are a bigger industry than movies and north american sports combined, thanks to the pandemic.* `https://www.marketwatch.com/story/videogames-are-a-bigger-industry-than-sports-and-movies-combined-thanks-to-the-pa` acesso em 26/04/2021. 1

[7] Lopes, Ricardo e Bidarra, Rafael: *Adaptivity challenges in games and simulations: A survey.* IEEE Transactions on Computational Intelligence and AI in Games, páginas 85–99, 2011. 1

[8] Novak, Jeannie: *Desenvolvimento de Games.* Cengage Learning, 2011, ISBN 9788522106325. 1, 9, 10, 16

[9] Schell, Jesse: *The Art of Game Design: A Book of Lenses.* AK Peters, 2a edição edição, 2014, ISBN 1466598646, 9781466598645. 1

[10] Chang, David Michael Jordan: *Dynamic difficulty adjustment in computer games.* Proceedings of the 11th Annual Interactive Multimedia Systems Conference, 2013. 1

[11] Gustavo Andrade, Geber Ramalho, Alex Sandro Gomes e Vincent Corruble: *Dynamic game balancing: an evaluation of user satisfaction.* American Association for Artificial Intelligence, 2006. 1

[12] Stenros, Jaakko: *The game definition game: A review.* Games and Culture, 2016. 4

[13] Juul, Jesper: *The game, the player, the world: Looking for a heart of gameness.* Level Up: Digital Games Research Conference Proceedings, páginas 30–45, 2003. 4

[14] Huizinga, Johan: *Homo Ludens: o jogo como elemento da cultura.* Perspectiva, 6ª edição edição, 2010, ISBN 8527300753, 9788527300759. 6, 9, 14

[15] Koster, Raph: *A Theory of Fun for Game Design.* Paraglyph Press, 1ª edição edição, ISBN 1449363210, 9781449363215. 6, 9

[16] Salen, Katie e Zimmerman, Eric: *Rules of Play: Game Design Fundamentals.* The MIT Press, 2003, ISBN 0262240459, 9780262240451. 6, 10

[17] Salen, Katie e Zimmerman, Eric: *The Game Design Reader: A Rules of Play Anthology.* The MIT Press, 2006, ISBN 0262195364, 9780262195362. 6

[18] Xavier, Guilherme: *Condição Eletrolúdica: Cultura Visual Nos Jogos Eletrônicos.* Editora Novas Idéias, 2010, ISBN 9788560284153. 6

[19] Prado, Gilberto e Vannucchi, Hélia: *Discutindo o conceito de gameplay.* Texto Digital, páginas 130–140, 2009. 6

[20] Mello, Vinícius e Perani, Letícia: *Gameplay x playability: defining concepts, tracing differences.* SBC - Proceedings of SBGames 2012, 2012. 6

[21] Crawford, Chris: *The art of computer game design.* Osborne/McGraw-Hill Berkley, 1984, ISBN 0881341177. 7

[22] Minkkinen, Toni: *Basics of plataform games.* Dissertação de Graduação, Kajaanin Ammattikorkeakoulu University of Applied Sciences, 2016. 8, 24

[23] Fullerton, Tracy: *Game Design Workshop: A Playcentric Approach to Creating Innovative Games.* A K Peters/CRC Press, 3ªª edição, 2014, ISBN 1482217163, 9781482217162. 9

[24] Aguiar, João Victor Santos e Fernandes, Lucas Vanderlei: *Ajuste dinâmico de dificuldade em jogos a partir de variáveis do jogo e do usuário.* Dissertação de Graduação, Universidade de Brasília, 2017. 9, 15

[25] Fortguno, Nick: *The strange case of the casual gamer.* Game Usability: Advice from the Experts for Advancing the Player Experiences, 2008. 10

[26] Csikszentmihalyi, Mihaly: *Flow: The Psychology of Optimal Experience.* New York: Harper Row, ISBN ISBN 0061339202, 9780061339202. 10, 11, 22

[27] Sherry, John L.: *Flow and media enjoyment.* Communication Theory, páginas 328 – 347, 2004. 12

[28] Chen, Jenova: *Flow in games (and everything else).* ACM magazine, 2007. 12

[29] Sweetser, Penelope e Wyeth, Peta: *Gameflow: A model for evaluating player enjoyment in games*. Computers in Entertainment, 2005. 12

[30] Aponte, Maria Virginia e Levieux, Guillaume: *Difficulty in video games: An experimental validation of a formal definition*. 2011. 14, 35

[31] Byrne, Ed: *Game Level Design*. Charles River Media, 1ª edição, ISBN 1584503696, 978-1584503699. 14

[32] *Difficulty in dexterity-based platform games*. `https://www.gamedev.net/articles/game-design/game-design-and-theory/difficulty-in-dexterity-based-platform-games-r2055/`. acesso em 04/03/2004. 15

[33] Lope, Javier de, Darío Maravall *et al*: *Adaptation, anticipation and rationality in natural and artificial systems: computational paradigms mimicking nature*. Natural Computing, 2009. 16

[34] Charles, Darryl, A Kerr M McNeill M McAlister M Black J Kcklich A Moore e K Stringer: *Player-centred game design: Player modelling and adaptive digital games*. Proceedings of the digital games research conference, 2005. 16

[35] Zohaib, Mohammad: *Dynamic difficulty adjustment (dda) in computer games: A review*. Advances in Human-Computer Interaction, 2018. 17, 18

[36] Gustavo Andrade, Geber Ramalho, Hugo Santana e Vincent Corruble: *Extending reinforcement learning to provide dynamic game balancing*. Proceedings of the Workshop on Reasoning, Representation, and Learning in Computer Games, 19th International Joint Conference on Artificial Intelligence (IJCAI), 2005. 17

[37] Hunicke, Robin: *The case for dynamic difficulty adjustment in games*. Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology, páginas 429–433, 2005. 19

[38] Parekh, Ravi: *Staying in the flow using procedural content generation and dynamic difficulty adjustment*. Dissertação de mestrado, Worcester Polytechnic Institute, 2017. 19

[39] Diana Lora, Antonio A. Sánchez-Ruiz, Pedro A. González Calero e Marco A. Gómez Martín: *Dynamic difficulty adjustment in tetris*. Proceedings of the Twenty-Ninth International Florida Artificial Intelligence Research Society Conference, 2015. 19

[40] Compton, Kate e Mateas, Michael: *Procedural level design for platform games*. American Association for Artificial Intelligence, 2006. 19, 20, 21

[41] Gillian Smith, Mee Cha e Jim Whitehead: *framework for analysis of 2d platformer levels*. Sandbox '08 Proceedings of the 2008 ACM SIGGRAPH symposium on Video games,, páginas 75–80, 2008. 20

[42] Sorenson, Nathan e Philippe Pasquier: *Towards a generic framework for automated video game level creation*. Applications of Evolutionary Computation, 2010. 21

[43] Koens, Erik: *Generating non-monotone 2d platform levels and predicting difficulty.* Dissertação de mestrado, Utrecht University, 2015. 30

Apêndice A

Questionário

Perguntas sobre dados sociodemográficos foram aplicadas antes do jogo iniciar e perguntas sobre o jogo ao final da sessão de teste. As opções de resposta de cada questão estão em colchetes, sendo essas: caixa de texto (resposta digitada), múltipla escolha (lista de opções em que uma deve ser escolhida), marcação (cada item pode ser marcado como verdadeiro ou falso), escala (lista de valores que representam extremos de uma afirmação, um deve ser escolhido) e grade de escalas (a mesma lista de valores aplicada a diferentes opções, em cada caso um valor deve ser escolhido).

A.1 Dados sociodemográficos

1. Qual a sua idade? [caixa de texto]
2. Qual o seu gênero? [múltipla escolha]
    - Feminino
    - Masculino
    - Outro
3. Quanto você se considera um jogador dedicado? [escala de 1 a 5]
    - Casual - Dedicado
4. Você está familiarizado com jogos de plataforma (ex: Mario, Sonic)? [escala de 1 a 5]
    - Pouco familiarizado - muito familiarizado
5. Qual sua relação com jogos de plataforma [escala de 1 a 5]
    - Não gosta - gosta muito

6. Em qual dificuldade você prefere jogar? [múltipla escolha]

    - Fácil
    - Médio
    - Difícil

7. Com que frequência você joga nos seguintes dispositivos? [grade de escalas de 1 a 5]

    - Nunca - Diariamente

    1. Celular / tablet
    2. Console de mesa (ex: PlayStation 4, XBOX One)
    3. Console portátil (ex: Nintendo 3DS, Playstation Vita)
    4. Computador pessoal (desktop ou laptop)

A.2 Jogo

1. Dentre as opções abaixo, quais foram as ações que considerou mais importantes para influenciar o desempenho que teve? [múltipla escolha]

    - Alcançar o final da fase
    - Coletar chaves
    - Dominar os controles da protagonista
    - Completar a fase o mais rápido possível
    - Derrotar o máximo de inimigos possível
    - Explorar ao máximo o ambiente
    - Evitar sofrer danos
    - Melhorar o desempenho próprio no jogo

2. Como você considera o seu desempenho no jogo? [escala de 1 a 5]

    - Péssimo - Ótimo

3. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 5]

    - Discordo muito - Concordo muito

    1. Minha atenção estava inteiramente focada no jogo

2. Em vários momentos me vi fazendo coisas automaticamente, sem que tivesse que pensar
3. Eu perdi a noção do tempo enquanto jogava o jogo

4. Em relação a cada opção, você concorda com a seguinte afirmação: o jogo se tornou mais desafiador com a presença desta característica? [grade de escalas de 1 a 5]

   - Discordo muito - Concordo muito

   1. Plataformas com maior distância vertical (precisa pular mais alto para alcançá-las)
   2. Plataformas com maior distância horizontal (precisa pular mais longe para alcançar - esquerda ou direita)
   3. Tamanho reduzido das plataformas (espaço menor para andar)

5. Você concorda com as seguintes afirmações? [grade de escalas de 1 a 5]

   - Discordo muito - Concordo muito

   1. O salto da personagem é difícil de controlar
   2. O salto da personagem é satisfatório e responsivo

6. Em relação a cada opção, você concorda com a seguinte afirmação: o jogo se tornou mais desafiador com a presença deste elemento? [grade de escalas de 1 a 5]

   - Discordo muito - Concordo muito

   1. Obstáculos (ex: pilares destrutíveis, caixas, etc.)
   2. Plataformas móveis com movimentação vertical
   3. Plataformas móveis com movimentação horizontal
   4. Gap (buracos necessários saltar para ultrapassar)

7. Em relação a cada opção, você concorda com a seguinte afirmação: o jogo se tornou mais desafiador com a presença deste elemento? [grade de escalas de 1 a 5]

   - Discordo muito - Concordo muito

   1. Ácido
   2. Espinhos
   3. Inimigos com ataque simples

4. Inimigos com ataque a distância

8. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 5]

   - Discordo muito - Concordo muito

   1. O jogo foi desafiador
   2. Eu fui desafiado pelo jogo, mas achei capaz de superar todos os desafios
   3. A dificuldade do jogo foi apropriada
   4. Eu gostei da experiência de jogar esse jogo