



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ajuste Dinâmico de Dificuldade pelo desempenho e perfil de jogador em jogo de plataforma

Marcos Paulo Cayres Rosa

Dissertação apresentada como requisito parcial para
conclusão do Mestrado em Informática

Orientador

Prof. Dr. Ricardo Pezzuol Jacobi

Coorientador

Prof.a Dr.a Carla Denise Castanho

Brasília
2022

Ficha Catalográfica de Teses e Dissertações

Esta página existe apenas para indicar onde a ficha catalográfica gerada para dissertações de mestrado e teses de doutorado defendidas na UnB. A Biblioteca Central é responsável pela ficha, mais informações nos sítios:

<http://www.bce.unb.br>

<http://www.bce.unb.br/elaboracao-de-fichas-catalograficas-de-teses-e-dissertacoes>

Esta página não deve ser incluída na versão final do texto.

Dedicatória

Dedico à minha mãe, aos meus avós, aos meus irmãos, ao meu padrasto, aos meus professores e aos meus amigos que percorreram essa jornada universitária comigo.

Agradecimentos

Agradeço a minha orientadora Carla Denise Castanho, ao meu orientador Ricardo Pezzuol Jacobi, ao professor Maurício Sarmet, aos professores e Tiago Silva e Esteban Clua por participarem da banca avaliadora e a todos os voluntários que participaram dos testes realizados.

Resumo

O Ajuste Dinâmico de Dificuldade (ADD) dos jogos pode desempenhar um papel importante para aumentar o envolvimento e a diversão do jogador. A dificuldade de jogo pode ser adaptada de acordo com o desempenho do jogador, seu estado afetivo ou usando um modelo híbrido que combina as duas abordagens. Ademais, pode adaptar configurações ou componentes do jogo e utilizar métricas pré-estabelecidas ou aprendizado de máquina para análise do que será adaptado. Este trabalho investiga os distintos mecanismos de um sistema de ADD para um jogo de plataforma adaptar adequadamente seu nível de dificuldade e manter o jogador em um estado de fluxo. Este trabalho contribui com a definição de um método que estima a dificuldade do jogo a partir de características específicas de componentes comuns ao gênero de plataforma. Também são revisadas métricas para medição do estado do fluxo e do perfil do jogador e propostas regras para criação de níveis ao testar modelos de ADD. O ajuste proposto varia o tamanho da plataforma e a altura do salto, comparando distintas abordagens a partir de sistemas do jogo e verificando a eficiência de cada uma em relação ao monitoramento e análise dos dados e ao controle da adaptação dos componentes. Um jogo de plataforma de código aberto foi adaptado para suportar os algoritmos de ADD e para execução de testes com grupos de amostra, nos quais os participantes respondiam a questionários e tiveram seus dados coletados para fins de investigação. Os resultados indicaram que a dificuldade de jogos de plataforma pode ser estimada pelos componentes dos níveis, incluindo correlação entre a dificuldade e os dados de desempenho dos jogadores. Além disso, perfis de jogadores foram previstos a partir de dados brutos da sessão do jogo e utilizados com métodos de aprendizado de máquina para definir a progressão de dificuldade. Por fim, os modelos de ADD foram capazes de ajustar a dificuldade do jogo aos jogadores, diminuindo a dispersão entre os dados de desempenho e mantendo o jogador em estado de fluxo, especialmente ao utilizar redes neurais diretas para predição da dificuldade experienciada e do perfil do jogador.

Palavras-chave: jogos eletrônicos; ajuste dinâmico de dificuldade; plataforma 2D; rede neural; perfil de jogador

Abstract

The Dynamic Difficulty Adjustment (DDA) of games can play an important role in increasing the player engagement and fun. Gameplay difficulty can be adapted according to the player's performance, its affective state or by using a hybrid model that combines both approaches. In addition, you can adapt game settings or components and use pre-established metrics or machine learning to analyze what will be adapted. This work investigates the different mechanisms of an DDA system for a platform game to adequately adapt its difficulty level and keep the player in a state of flow. This work contributes with the definition of a method that estimates the game's difficulty based on specific characteristics of components common to the platform genre. Metrics for measuring the flow state and player profile are also reviewed, and rules for creating levels when testing ADD models are proposed. The proposed adjustment varies the size of the platform and the height of the jump, comparing different approaches from the game systems and verifying the efficiency of each one in relation to the monitoring and analysis of the data and the control of the components adaptation. An open source platform game was adapted to support the ADD algorithms and to run tests with sample groups, in which participants answered questionnaires and had their data collected for research purposes. The results indicated that the difficulty of platform games can be estimated by the components of the levels, including correlation between the difficulty and player performance data. In addition, player profiles were predicted from raw game session data and used with machine learning methods to define difficulty progression. Finally, the DDA models were able to adjust the game difficulty to the players, decreasing the dispersion between the performance data and keeping the player in a state of flow, especially when using feedforward neural networks to predict the difficulty experienced and the player's profile.

Keywords: video game, dynamic difficult adjustment, 2D platform; neural network; player profile

Sumário

1	Introdução	1
1.1	Definição do problema	1
1.2	Objetivos	2
1.3	Proposta	3
1.4	Estruturação	4
2	Fundamentação Teórica	5
2.1	Jogo	5
2.1.1	Taxonomia de jogos	6
2.1.2	Categorização de jogos de plataforma	7
2.2	Jogador	9
2.2.1	Classificação de jogadores	9
2.2.2	Emoção do jogador	10
2.2.3	Afetividade	12
2.2.4	A experiência do jogador	12
2.3	Fluxo	15
2.3.1	Fluxo em jogos	16
2.4	Dificuldade em jogos	17
2.4.1	Adaptatividade	19
2.4.2	Ajuste Dinâmico de Dificuldade	21
2.4.3	Dificuldade em jogos de plataforma	23
2.5	Aprendizado de Máquina	25
2.5.1	Rede Neural Artificial	26
3	Trabalhos Correlatos	28
3.1	Avaliação da dificuldade	28
3.1.1	<i>Design</i> do desafio	28
3.1.2	Modelo estatístico	29
3.1.3	Medição da dificuldade de componentes individuais	29

3.2	Inteligência Artificial	31
3.2.1	Aplicação de Inteligência Artificial em jogos	31
3.2.2	Adaptatividade com Aprendizado de Máquina	33
3.3	ADD a partir do desempenho do jogador	35
3.3.1	Estado da Arte	36
3.3.2	Modelagem do jogador	37
3.3.3	Distribuição de recursos e desafios	38
3.3.4	Compatibilidade do desafio	38
3.3.5	Perfil do jogador	39
3.4	ADD afetivo	40
3.4.1	Estado da arte	41
3.4.2	Análise em Tempo Real	42
3.4.3	Afetividade associada ao desempenho do jogador	43
3.5	ADD em jogos de plataforma	44
3.5.1	Geração procedural de níveis	44
3.5.2	Dificuldade associada à geração dinâmica adaptativa	46
3.5.3	Abordagens para personalização de níveis	47
3.5.4	Dados fisiológicos na adaptação de jogos de plataforma	48
4	Contextualização	49
4.1	Propostas em estudos anteriores	49
4.2	Medição de dificuldade	50
4.2.1	Implementação da estimativa da dificuldade	52
4.3	Dados afetivos	55
4.4	ADD híbrido	56
4.4.1	Extração de dados	59
4.4.2	Análise de dados de desempenho	62
4.4.3	Adaptações a partir da dificuldade	64
4.5	O jogo	67
4.5.1	Motor de jogo	68
4.5.2	Versão original do jogo	69
4.5.3	Modificações do jogo	70
4.6	Análises complementares	75
4.6.1	Balanceamento e verificação de medição de dificuldade	78
4.6.2	Comparação entre modelos por desempenho, afetivo e híbrido	81
4.6.3	Comparação entre adaptação de plataforma, salto e combinado	87

5	Proposta do Ajuste Dinâmico de Dificuldade	90
5.1	Proposta	90
5.2	Aprimoramento do método	92
5.2.1	Modificações do Sistema de Controle	93
5.2.2	Atualização do método de medição de dificuldade de plataformas	95
5.2.3	Atualização do método de medição de dificuldade do nível	102
5.2.4	Modificações do Sistema de Análise	104
5.3	Modelo de Aprendizado de Máquina	109
5.3.1	Adaptação por dificuldade experienciada e perfil de jogador	110
5.4	Modificações adicionais do jogo	114
6	Testes e Resultados	117
6.1	Testes	117
6.1.1	Descrição dos níveis	120
6.2	Averiguação da estimação de dificuldade e dos dados de jogadores	124
6.3	Comparação entre modelos do Sistema de Controle do ADD	132
6.4	Comparação entre modelos do Sistema de Análise do ADD	143
7	Considerações Finais	156
7.1	Principais contribuições	156
7.2	Conclusões	158
7.3	Trabalhos futuros	159
	Referências	161
	Apêndice	169
A	Questionário	170
A.1	Dados Sociodemográficos	170
A.2	Primeira bateria de testes	172
A.2.1	Nível	172
A.2.2	Jogo	172
A.3	Segunda e terceira bateria de testes	174
A.3.1	Jogo	174

Lista de Figuras

2.1	Imagens dos jogos <i>Super Mario Bros.</i> (A), <i>Little Big Planet</i> (B) e <i>Sonic the Hedgehog</i> (C) [1].	8
2.2	Modelo Circumplexo de Afeto com as emoções geradas pela combinação de excitação e valência (adaptado de [2]).	11
2.3	Tipos de usuário Hexad [3].	14
2.4	a) Experiência do estado de fluxo em um jogador; b) Jogador ao encontrar entropias psíquicas; c) Diferentes jogadores e zonas de fluxo (adaptado de [4]).	17
2.5	Diagrama de blocos de um jogo afetivo adaptativo (adaptado de [5]).	21
2.6	Ciclo de ADD orientado pelo sistema (adaptado de [4]).	22
2.7	Elementos de design para o jogo <i>Super Mario Bros.</i> , em sentido horário: bloco, cano, buraco, inimigo, plataforma e escadaria [6].	24
2.8	Ilustração do modelo conceitual de estrutura de níveis (adaptado de [7]).	25
3.1	Exemplo de um grafo representante de um nível [8]. As arestas amarelas representam as rotas através do nível e as arestas brancas representam arestas arbitrárias.	30
3.2	Configuração de jogo de plataforma para geração de conteúdo de jogo personalizado com base na classificação de expressões faciais [9].	32
3.3	Configuração experimental para um jogo de treinamento de memória com um robô Nao movendo seus braços para gerar estímulos ao tocar em objetos inteligentes [10].	33
3.4	Habilidade e perfil de risco do jogador medidos dinamicamente para identificar tamanho do filtro de partícula e limiar de decisão [11].	40
3.5	Exemplo de um nível com dois segmentos independentes, ajustado individualmente para dois jogadores de diferentes habilidades [12].	46
3.6	Captura de tela do jogo protótipo usado em [12].	47
4.1	Representações de desafios em jogos de plataforma (adaptado de [12]).	51
4.2	Representação gráfica do conceito de margem de erro [12].	54

4.3	(a) Classificação do salto de acordo com a posição de destino (P_1) em relação à origem (P_0); (b) Valores de correção máximo e mínimo para tornar possível o salto de uma dada trajetória (adaptado de [12]).	55
4.4	Exemplo da variação da amplitude GSR ao longo do tempo ¹	57
4.5	Diagrama cíclico do Ajuste Dinâmico de Dificuldade híbrido orientado pelo sistema. Os sistemas estão em verde, enquanto as etapas relacionadas ao desempenho do jogador estão em vermelho e à afetividade, em amarelo. . .	58
4.6	Representações dos tipos de plataforma no jogo. As setas indicam a origem e o destino do salto, com a plataforma correspondente possuindo sua nomenclatura associada.	61
4.7	Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0,1 e (c) 0,9, respectivamente.	66
4.8	Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0, respectivamente.	66
4.9	Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0,7 e (c) 1, respectivamente.	66
4.10	Captura de tela do mesmo segmento com as diferentes alturas máximas do salto. (a) Amplitude máxima sem ADD aplicado; (b) Com ADD de salto aplicado para o valor de dificuldade global mais fácil; e (c) Com ADD de salto aplicado para o valor de dificuldade global mais difícil.	67
4.11	Captura de tela da versão original do jogo <i>The Explorer: 2D</i>	68
4.12	Exemplificação da HUD na versão original do jogo, contendo vidas, inventário e interface de diálogo.	70
4.13	Arquivos de entrada para geração do nível.	71
4.14	Níveis gerados a partir dos arquivos de entrada.	72
4.15	Ilustração dos principais componentes do jogo, conforme é apresentada ao usuário pelo menu inicial e pelo de pausa.	73
4.16	Exemplificação da modelagem de ladrilhos das plataformas.	74
4.17	No sentido horário, interface dos objetivos, tela inicial, questionário e menu de pausa.	74
4.18	Comparação entre a média das dificuldades medidas pelo questionário e a dificuldade estimada pelo algoritmo desenvolvido em cada nível da primeira bateria de testes de Rosa (adaptado de [13]).	78

4.19	Modelo de regressão para o tempo jogado como variável preditora (X) e dificuldade experienciada como variável de resposta (Y) a partir de dados da primeira bateria de testes de Rosa.	81
4.20	Valor médio e desvio padrão das dificuldades medidas pelo questionário em cada nível da segunda bateria de testes de Rosa, dividido nos três conjuntos de dificuldade testados e agrupados pelos quatro modelos (adaptado de [13]).	82
4.21	Diagrama de caixa para o tempo jogado, respectivamente, para os conjuntos de níveis (a) fácil, (b) normal e (c) difícil, comparando os quatro modelos na segunda bateria de testes de Rosa.	84
4.22	Frequência de respostas sobre a dificuldade ser mais adequada no nível atual ou anterior com frequências agrupadas por conjuntos de dificuldade, questionadas ao final dos níveis da segunda bateria de testes de Rosa (adaptado de [13]).	85
4.23	Resultados do agrupamento <i>K-means</i> com 3 <i>clusters</i> para (a) primeira e (b) segunda baterias de testes de Rosa com quantidade de saltos falhos e tempo jogado ou quantidade de saltos e tempo jogado dividido pela quantidade de saltos como parâmetros, respectivamente.	86
4.24	Valor médio e desvio padrão das questões relacionadas ao estado de fluxo na bateria de testes de Santos e Moraes, agrupadas pelos quatro tipos de adaptação.	87
4.25	Valor médio e desvio padrão das questões relacionadas à adequação ou dificuldade na bateria de testes de Santos e Moraes, agrupadas pelos quatro tipos de adaptação.	87
5.1	Diagrama cíclico completo do Ajuste Dinâmico de Dificuldade orientado pelo sistema. Os sistemas estão definidos em verde, as ações referentes ao desempenho do jogador em laranja e a afetividade em amarelo, as diferentes opções de análise em tons de azul e as diferentes opções de adaptação em tons de cinza.	92
5.2	Exemplificação dos tipos de plataforma, em ordem horária: Inferior Direita, Inferior Esquerda, Superior Esquerda e Superior Direita. Estão indicados origens, destinos e caminhos a serem percorridos por setas. Os destinos possuem salvas as informações da sua primeira e última posição.	96
5.3	Ilustração de plataforma móvel, com informações da primeira e última posição da plataforma e do caminho a ser percorrido. Nota-se que a posição final varia de acordo com o caminho.	96

5.4	Representação gráfica do conceito de margem de erro para saltos em que a posição final da plataforma é menor do que a posição final que o salto terminaria.	97
5.5	Representação gráfica do conceito de margem de erro para saltos em que a origem está acima do destino (Inferior).	99
5.6	Representação gráfica do conceito de margem de erro para saltos em que a origem vem antes do destino (Esquerda), sendo (a) para casos que o destino vem acima da origem (Superior Esquerda) e (b) para casos que o destino vem abaixo da origem (Inferior Esquerda).	100
5.7	Ilustração da diferença de dificuldade para amplitude do salto, sendo que (a) amplitude mínima do salto está representada pela parábola de cor vermelha (mais difícil), (b) média na cor branca e (c) máxima na cor verde (mais fácil).	100
5.8	Exemplificação de Rede Neural Artificial.	112
5.9	Captura de tela da interface do jogo em versão de aplicativo Android.	114
5.10	Captura de tela do nível de tutorial.	115
6.1	Captura de tela de trecho do nível 4 da primeira bateria de testes, demonstrando dificuldade gerada a partir do <i>design</i> do nível.	121
6.2	Captura de tela de trecho do nível 7 da primeira bateria de testes, demonstrando nível com dificuldade gerada a partir de componentes de plataforma com alta dificuldade individual e baixa densidade.	122
6.3	Captura de tela de trecho do nível 8 da terceira bateria de testes.	122
6.4	Captura de tela de trecho do nível 8 da terceira bateria de testes com adaptação para facilitar o nível ao aplicar dificuldade global de 0,2.	123
6.5	Diagrama de caixa com as médias das dificuldades experienciadas pelos jogadores, medidas por meio do questionário, e os valores estimados por algoritmos na primeira bateria de testes. A estimacão de dificuldade pelo método atualizado está em amarelo e a estimacão pelo método anterior está normalizada e em vermelho.	124
6.6	Diagrama de caixa com as médias das dificuldades experienciadas pelos jogadores, medidas por meio do questionário e separadas por perfil de jogador predominante (não Revolucionários e Revolucionários) na primeira bateria de testes.	126
6.7	Média e desvio padrão das medições de diversão, desafio, frustraão e tédio dos participantes na primeira bateria de testes.	126

6.8	Média e desvio padrão das medições de desempenho próprio, perfil de risco e de resiliência dos participantes na primeira bateria de testes, divididas por perfil de jogador.	128
6.9	Média e desvio padrão do desafio proporcionado por componentes específicos, ordenados de forma decrescente, de acordo com respostas dos participantes na primeira bateria de testes, divididas por perfil de jogador.	129
6.10	Média e desvio padrão das atividade do jogo, ordenadas de forma decrescente, que influenciaram o desempenho dos participantes na primeira bateria de testes, divididas por perfil de jogador.	130
6.11	Diagrama de caixa com a porcentagem de chaves coletadas, entre 0 e 1, em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.	131
6.12	Diagrama de caixa com o tempo andando em direção oposta, em segundos, ao final em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.	131
6.13	Diagrama de caixa com a quantidade de ataques a distância em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.	132
6.14	Diagrama de caixa com a porcentagem de saltos de sucesso, entre 0 e 1, em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.	133
6.15	Diagrama de caixa com o tempo jogado, em segundos, em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.	133
6.16	Diagrama de caixa com a quantidade de mortes em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.	134
6.17	Diagrama de caixa com a porcentagem de saltos de sucesso, entre 0 e 1, em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.	134
6.18	Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.	135
6.19	Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.	135

6.20	Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes com predileção pela dificuldade fácil na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.	136
6.21	Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes com predileção pela dificuldade fácil na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.	137
6.22	Média, desvio padrão, mínimo e máximo da dificuldade de cada nível ao adaptar para os participantes da segunda bateria de testes, divididas por modelo de ADD.	137
6.23	Média, desvio padrão, mínimo e máximo da dificuldade global (a qual o nível deveria seguir) para os participantes da segunda bateria de testes, divididas por modelo de ADD.	138
6.24	Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na segunda bateria de testes, divididas por modelo de ADD.	139
6.25	Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes com predileção pela dificuldade fácil na segunda bateria de testes, divididas por modelo de ADD.	139
6.26	Média e desvio padrão das respostas referentes ao desafio proporcionado pelo jogo aos participantes na segunda bateria de testes, divididas por modelo de ADD.	140
6.27	Média e desvio padrão das respostas referentes a sentimentos experienciados no jogo pelos participantes na segunda bateria de testes, divididas por modelo de ADD.	141
6.28	Média e desvio padrão das respostas referentes a dificuldade percebida no jogo pelos participantes na segunda bateria de testes, divididas por modelo de ADD.	141
6.29	Média e desvio padrão das respostas referentes a dificuldade percebida no jogo pelos participantes com predileção pela dificuldade fácil na segunda bateria de testes, divididas por modelo de ADD.	142
6.30	Média e desvio padrão das respostas referentes ao estado de fluxo experienciado pelos participantes na segunda bateria de testes, divididas por modelo de ADD.	142
6.31	Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.	144

6.32	Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.	144
6.33	Diagrama de caixa com a quantidade de saltos falhos em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.	145
6.34	Diagrama de caixa com a porcentagem de saltos de sucesso em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.	145
6.35	Diagrama de caixa com a porcentagem de chaves coletadas em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.	146
6.36	Diagrama de caixa com a porcentagem de saltos de sucesso em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Controle do ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.	146
6.37	Diagrama de caixa com a porcentagem de chaves coletadas em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Controle do ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.	147
6.38	Média, desvio padrão, mínimo e máximo da dificuldade de cada nível ao adaptar para os participantes da terceira bateria de testes, divididas por modelo de ADD.	148
6.39	Média, desvio padrão, mínimo e máximo da dificuldade global (a qual o nível deveria seguir) para os participantes da terceira bateria de testes, divididas por modelo de ADD.	149
6.40	Média e desvio padrão de cada nível ao adaptar para os participantes da terceira bateria de testes, divididas por modelo de ADD pela cor e pelo perfil de jogador predominante entre Espírito Livre e Realizador pelo tracejado das linhas.	149
6.41	Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na terceira bateria de testes, divididas por modelo de ADD.	150
6.42	Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.	151

6.43	Diagrama de caixa das respostas referentes a dificuldade percebida no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD.	151
6.44	Diagrama de caixa das respostas referentes a dificuldade percebida no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.	152
6.45	Diagrama de caixa das respostas referentes ao desafio proporcionado pelo jogo aos participantes na terceira bateria de testes, divididas por modelo de ADD.	153
6.46	Diagrama de caixa das respostas referentes ao desafio proporcionado pelo jogo aos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.	153
6.47	Diagrama de caixa das respostas referentes a sentimentos experienciados no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD.	154
6.48	Diagrama de caixa das respostas referentes ao estado de fluxo experienciado pelos participantes na terceira bateria de testes, divididas por modelo de ADD.	154

Lista de Tabelas

2.1 Exemplos de taxonomia dos jogos eletrônicos (adaptada de [14]).	7
2.2 Classificação de subgêneros de jogos de plataforma conforme a ênfase no movimento, nos confrontos e nas interações (adaptada de [12]).	9
4.1 Porcentagem de participantes que concluíram, tempo mínimo, tempo máximo, tempo médio e desvio padrão relacionados à permanência em cada nível da primeira bateria de testes de Rosa (adaptado de [13]).	79
4.2 Coeficiente de determinação (R^2) encontrado para cada uma das variáveis preditoras declaradas (X), considerando a dificuldade como a variável de resposta (Y). Sem transformação logarítmica, com transformação logarítmica e com transformação logarítmica desconsiderando níveis que não foram completados pelo jogador na primeira bateria de testes de Rosa.	80
4.3 Porcentagem de participantes que concluíram, tempo mínimo, tempo máximo, tempo médio e desvio padrão relacionados à permanência em cada nível da segunda bateria de testes de Rosa (adaptado de [13]).	83
4.4 Valor-p medido a partir do teste de Friedman, de acordo com os dados de desempenho para cada conjunto de níveis por dificuldade para segunda bateria de testes de Rosa.	83
4.5 Valor de Hopkins encontrado para distintos métodos (<i>K-means</i> , <i>Mean-Shift</i> e <i>Aglomerativo</i>), de acordo com um subconjunto de dados de desempenho capturados na segunda bateria de testes de Rosa.	86
4.6 Média, mediana e desvio padrão do tempo para completar o jogo em cada adaptação na bateria de testes de Santos e Moraes.	88
4.7 Média, mediana e desvio padrão do tempo para completar o jogo em cada adaptação com jogadores que preferem a dificuldade fácil na bateria de testes de Santos e Moraes.	89
4.8 Média, mediana e desvio padrão da quantidade de mortes por queda em cada adaptação com jogadores que preferem a dificuldade fácil na bateria de testes de Santos e Moraes.	89

6.1 Porcentagem de participantes que completaram cada nível com as correspondentes dificuldades estimadas, a média das dificuldades medidas e o valor-p da diferença entre os valores de dificuldade na primeira bateria de testes. . . 125

Lista de Abreviaturas e Siglas

ACM *Association for Computing Machinery.*

ADD Ajuste Dinâmico de Dificuldade.

BFS Busca em Largura.

BVP Pulso de Volume de Sangue.

ECG Eletrocardiografia.

EDA Atividade Eletrodérmica.

EEG Eletroencefalograma.

EMG Eletromiografia.

FPS *First Person Shooter.*

GCP Geração de Conteúdo Procedural.

GSR Resposta Galvânica da Pele.

HUD Tela de Alerta.

IA Inteligência Artificial.

ICG Cardiografia de Impedância.

MCTS Busca em Árvore Monte Carlo.

NPC Personagem Não Jogável.

PPG Fotopletismografia.

RNA Rede Neural Artificial.

SNA Sistema Nervoso Autônomo.

SNC Sistema Nervoso Central.

SO Sistema Operacional.

TCLE Termo de Consentimento Livre e Esclarecido.

Capítulo 1

Introdução

1.1 Definição do problema

A indústria de jogos cresceu significativamente durante as últimas décadas, fazendo com que os desenvolvedores de jogos eletrônicos tenham acesso a um vasto público para vender seus produtos. Nesse sentido, os requisitos e regras para fazer bons jogos precisam se adequar aos novos cenários, criando jogos divertidos para cada perfil de consumidor. Novos mecanismos são exigidos para abranger uma diversidade de jogadores e permitir direcionar os jogos a usuários com distintos níveis de habilidades e preferências.

Sistemas adaptativos buscam fornecer ao jogador uma experiência personalizada que ajuste o jogo mediante objetivos identificáveis, mensuráveis e influenciáveis [15]. No caso, procura-se alcançar o equilíbrio do jogo para distintos jogadores, de forma consistente, justa e divertida [16]. Com isso em vista, o jogo pode manter o jogador em um estado de fluxo, oferecendo um desafio contínuo, em conformidade com a habilidade do jogador e que o deixe motivado a continuar jogando [17].

Tédio ou frustração podem ser causados por jogos não-adaptativos ao não avaliar a correspondência entre o grau do desafio e o perfil do jogador [18]. Em contraponto aos níveis de dificuldade pré-estabelecidos, existe o Ajuste Dinâmico de Dificuldade (ADD), o qual é uma técnica emergente que busca adaptar a dificuldade de um jogo durante a sua execução de acordo com as habilidades de cada jogador [19].

O Ajuste Dinâmico de Dificuldade pode ser visto como um ciclo contendo 4 sistemas interligados [4][5]:

- Dados brutos do Jogador;
- O Monitoramento dos dados;
- A Análise dos dados para definir o que será ajustado; e

- O Controle dos ajustes no jogo.

Cada sistema pode ser dividido de acordo com suas características e a proposta dos ajustes. Assim, o Monitoramento (com sua Análise associada) pode ser feito baseado em dados de desempenho do jogador (a partir de variáveis do jogo), nos dados fisiológicos do jogador (capturados a partir de sensores) ou na combinação desses, assim caracterizando um ADD híbrido. Por sua vez, a Análise dos dados pode ser feita a partir de funções heurísticas pré-definidas ou por aprendizado de máquina e o Controle dos ajustes pode adaptar distintos fatores, como configurações do avatar, adaptações de conteúdos do mundo do jogo ou uma combinação de adaptações.

O foco da maioria dos estudos da área de ADD está relacionado à dificuldade percebida pelo jogador, avaliando isso por meio de questionários e dados de desempenho. Esses estudos demonstraram a eficiência do ADD em manter o jogador em estado de fluxo e levavam em consideração somente os dados de variáveis do jogo [20] [21] [22] [23] [24] ou as respostas fisiológicas do jogador [25] [26] [27] [28]. Observou-se que o uso de sensores possui uma menor proliferação, especialmente se a adaptação é realizada em tempo real.

Por sua vez, também há poucos estudos nos quais as métricas do estado de fluxo e as regras para adaptação tenham definições robustas em relação as definições do conceito de fluxo e do padrão de sistemas do jogo. Além disso, poucos casos aplicam aprendizado de máquina fora do contexto de adaptação de Personagem Não Jogável [29] ou associam a adaptação com características do jogador além de sua habilidade [11], como características específicas de perfis de jogadores. Ou seja, são escassas investigações aprofundadas sobre a eficiência do uso de sistemas híbridos, de ajustes que combinem e comparem distintos métodos em cada sistema do jogo e de métricas robustas para medição comparativa do estado de fluxo, assim como a expansão para outros gêneros de jogos.

1.2 Objetivos

Neste trabalho, são continuados os estudos de Rosa [13] e de Santos e Moraes [30] em um jogo do gênero plataforma. Esses trabalhos comparavam, respectivamente, os modelos de Monitoramento e Análise do Ajuste Dinâmico de Dificuldade (por desempenho, afetivo e híbrido) e distintas adaptações no Sistema de Controle (configuração da amplitude do salto do jogador, adaptações de componentes de plataforma e a combinação destes). Verificou-se a capacidade do mecanismo híbrido e da combinação de distintas adaptações em adequar a dificuldade ao jogador e mantê-lo em estado de fluxo.

Para o presente trabalho, objetiva-se aumentar a robustez das métricas de medição de dificuldade e de análise de componentes, generalizando para utilização em outros jogos de plataforma. Para construção do modelo proposto, o balanceamento do jogo será analisado

em relação a componentes genéricos de jogos de plataforma e às dificuldades correspondentes, também averiguando a eficiência de métodos computacionais para estimativa da dificuldade de componentes individuais e de níveis de jogos. Assim, a dificuldade estimada será comparada com a que foi experienciada pelos jogadores, permitindo uma melhor avaliação do ADD em todas as variações analisadas.

A partir disso, pretende-se definir regras para criação de níveis e para execução de adaptações, almejando implementar um sistema de ADD que possa ser aplicado para distintos formatos no Sistema de Controle e de Análise, separadamente. Com isso, os modelos serão comparados, averiguando quais possuem resultados superiores comparativamente e se todos são significativamente melhores do que o jogo sem a aplicação de ajustes. Para isso, será avaliado o estado de fluxo e a adequação da dificuldade, de acordo com respostas de questionários e com a dispersão de dados brutos do jogo captados durante sessões de teste.

1.3 Proposta

Considerando os objetivos apresentados, será adaptado um jogo de plataforma 2D de acordo com os componentes mais comuns a esse gênero. O jogo se assimila principalmente ao *Super Mario Bros.*, com o jogador controlando um avatar que pode se movimentar para frente ou para trás, pular ou cair, almejando percorrer um caminho até um ponto final enquanto supera obstáculos, enfrenta inimigos e obtém colecionáveis.

Dentre os 3 elementos principais do gênero de plataforma [12], o enfoque para as adaptações estará no movimento do avatar, em detrimento ao confronto com oponentes e a interação com o ambiente. Ou seja, considerando sua natureza mais determinista [8], o estudo prioriza componentes de plataformas e configurações do salto do jogador. Assim, o jogo será adaptado de maneira a progredir de acordo com as dificuldades experienciadas para cada jogador, procurando mantê-lo em uma faixa adequada de dificuldade.

O jogo possuirá um ADD que analisa o desempenho do jogador por funções heurísticas ou pelo aprendizado de máquina, relacionando a progressão de dificuldade com o perfil do jogador. A medição da dificuldade de um componente de plataforma será baseada nas fórmulas matemáticas propostas por Mourato [1], definida por meio da probabilidade de sucesso para executar um salto. O salto corresponde a uma parábola e a probabilidade de sucesso advém do erro horizontal e vertical da parábola de um ponto de origem em relação a um determinado ponto final.

Este trabalho irá apresentar a definição de métricas mais robustas para medição do estado de fluxo e do perfil de jogador, a partir da revisão da literatura e de análises adicionais das respostas dos participantes dos testes de Rosa [13] e de Santos e Moraes

[30]. Nesse sentido, serão propostas novas regras para definição de dificuldade a partir de componentes genéricos do gêneros de plataforma, incluindo a agregação de novas considerações para o método de Mourato [1], e para criação de níveis para teste de modelos de ADD, incluindo a separação entre sistemas do jogo para execução de testes. Outras contribuições do estudo incluem a correlação entre dados brutos do jogo com a dificuldade experienciada e o perfil do jogador, além da utilização de características relacionadas à dificuldade a partir de perfis de jogadores para melhoria do método de ADD.

Testes serão realizados para verificar o método de estimação de dificuldade de jogos de plataforma e correlação entre dados brutos da sessão de jogo com a dificuldade experienciada e os perfis de jogadores. Em um segundo momento, testes irão avaliar o ADD aplicado para o tamanho de componentes de plataforma e para configuração da amplitude do salto no Sistema de Controle, verificando as métricas para medição do estado de fluxo e para adequação da dificuldade em estudos com aplicação de ADD. Por fim, testes irão averiguar o uso de aprendizado de máquina e da associação de perfis de jogador com a dificuldade experienciada para melhor adequação do jogo e para manutenção do estado de fluxo pelo Sistema de Análise. Nesse cenário, será feita a comparação entre o ADD com funções heurísticas e com rede neural direta, sendo que a validação dos testes será realizada por meio da análise de dados brutos das sessões de jogo e de questionários.

1.4 Estruturação

Este trabalho foi estruturado da seguinte forma:

- **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 alguns trabalhos correlatos relevantes no contexto desta investigação.
- **Capítulo 4:** contextualiza o problema e relata os estudos prévios que se está dando continuidade.
- **Capítulo 5:** detalha a proposta do trabalho e as soluções empregadas.
- **Capítulo 6:** relata os testes conduzidos, os resultados obtidos e as respectivas análises.
- **Capítulo 7:** apresenta as conclusões do presente trabalho e sugestões para trabalhos futuros na área.

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 os sensores para medição da afetividade, o fluxo, além do ajuste de dificuldade de jogos eletrônicos.

2.1 Jogo

O conceito de jogo foi analisado por diversos autores por impactar uma ampla área de estudo. Mais de 60 definições foram feitas desde 1930, conforme estudo de 2017 [31]. Algumas definições possuem destaque no meio acadêmico, mesmo com a discordância entre distintos autores, como a de Juul [32]. Nessa definição, um jogo é um sistema formal baseado em regras com um resultado variável e quantificável. Em complemento, Juul apresenta 6 pontos relativos ao conceito de jogo:

1. Regras;
2. Resultado variável e quantificável;
3. Valores designados a possíveis resultados, evidenciando o objetivo que o jogador deve almejar;
4. Esforço do jogador visando influenciar o resultado;
5. Jogador ligado ao resultado;
6. Consequências da atividade opcionais negociáveis.

A ligação emocional do jogador é especialmente importante, pois influencia na resposta positiva ou negativa que o jogador pode ter a partir do resultado obtido no jogo. Nesse

sentido, um ponto essencial ao jogo é a diversão [33], se relacionando ao esforço do usuário e aos resultados por ele alcançados. A diversão nos jogos advém da compreensão e do domínio de padrões dos desafios do jogo, de acordo com Koester [34]. Por outro lado, o tédio pode surgir da necessidade de absorver novos padrões, quando o jogo não é mais capaz de fornecer variações ou a dificuldade não é condizente às habilidades do jogador.

O meio físico do jogo é um componente importante em jogos eletrônicos, porém esse não constitui o sistema por inteiro [35]. Ou seja, devem ser entendidos como um componente em um sistema maior, sendo estipulados quatro traços para resumir as qualidades especiais de um jogo eletrônico:

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

Salen e Zimmerman definem *gameplay* como um processo reativo, delimitado por regras e emergente de sequências de ações pertencentes a um ou mais jogadores [36]. Nesse contexto, o *gameplay* é como um espectro de possíveis ações e reações geradas pelo jogador e pelo jogo, sendo o ponto de encontro desses [37].

Por sua vez, Xavier aponta a jogabilidade (*playability*) como uma decorrência natural do lidar com o jogo [38]. Isto é, uma característica intrínseca ao conjunto de ações que são esperados do jogador com todos os componentes, sejam eles audiovisuais, emocionais ou cognitivos.

Logo, a jogabilidade pode se referir a como o jogador interage com o jogo, sendo afetada pela entrada, como o teclado ou controle. Enquanto isso, Mello indica que o *gameplay* pode ser visto como indiferente a entrada e se associar às mecânicas do jogo, sendo o resultado da interação entre a jogabilidade e as regras e possuindo grande influência sobre a experiência do jogador [39].

2.1.1 Taxonomia de jogos

A classificação taxonômica dos jogos serve como um guia para os estudos específicos de gênero e apresenta princípios fundamentais do design de jogos, incluindo similaridades e diferenças entre famílias de jogos e entre membros de cada família, como definido por Crawford [40]. Entretanto, classificações não são absolutas e existe discordância entre autores, pois há diversas definições de taxonomia para distintos aspectos do jogo e as taxonomias precisam refletir as evoluções na área de jogos. Um conjunto de taxonomias

está exemplificado na Tabela 2.1, com base na jogabilidade, no gênero e na quantidade de jogadores.

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, Civilization
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

Tabela 2.1: Exemplos de taxonomia dos jogos eletrônicos (adaptada de [14]).

2.1.2 Categorização de jogos de plataforma

Um jogo do gênero plataforma, como definido por Minkkinen [41], tem como cerne um personagem controlado pelo jogador, chamado avatar, que corre e pula para evitar obstáculos e/ou derrotar inimigos. Ademais, existem 3 aspectos principais para analisar jogos desse gênero, baseados nas mecânicas mais comuns [12]:

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 com o 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.

Considerando os aspectos e as funcionalidades identificados nos principais jogos do gênero, exemplificado na Figura 2.1, Mourato propôs a definição dos seguintes subgêneros para jogos de plataforma:

- 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 quebra-cabeça: 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 de 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 focar nenhum desses aspectos em particular. Com isso, cria um mecanismo de jogo mais complexo, mas mantém o equilíbrio entre esses recursos. Exemplo: *Little Big Planet*.

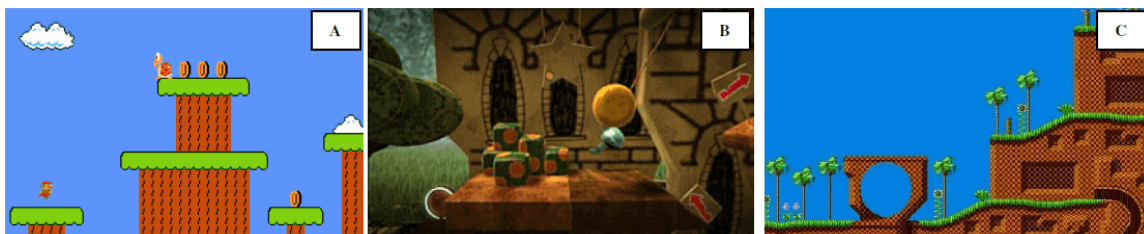


Figura 2.1: Imagens dos jogos *Super Mario Bros.* (A), *Little Big Planet* (B) e *Sonic the Hedgehog* (C) [1].

A Tabela 2.2 sumariza os subgêneros de acordo com a ênfase nas características consideradas.

Alguns jogos não podem ser classificados como um único subgênero, sendo projetados como híbridos. Além disso, caso o conteúdo do jogo seja decomposto em segmentos nos quais a jogabilidade é diferenciada significativamente, conseqüentemente não se encaixarão em uma única categoria, possuindo níveis de plataforma dentro de seu domínio de jogo [12]. Por exemplo, os jogos *The Adventures of Tintin* e *Another World* possuem estrutura de plataforma e outros gêneros (corrida e quebra-cabeça) durante sua narrativa.

Movimento	Confrontos	Interações	Subgênero
Fraco	Fraco	Fraco	Plataforma minimalista
Fraco	Fraco	Forte	Plataforma clássica
Fraco	Forte	Fraco	Plataforma de ação
Fraco	Forte	Forte	Não é plataforma
Forte	Fraco	Fraco	Plataforma clássica
Forte	Fraco	Forte	Plataforma de quebra-cabeça
Forte	Forte	Fraco	Plataforma de ação
Forte	Forte	Forte	Plataforma de compostos

Tabela 2.2: Classificação de subgêneros de jogos de plataforma conforme a ênfase no movimento, nos confrontos e nas interações (adaptada de [12]).

2.2 Jogador

O jogador pode ser visto como um agente que busca diversão no jogo [33], sendo preciso aceitar voluntariamente as regras e as restrições do jogo [42]. Ou seja, o jogador pode ser motivado a jogar pela busca de diversão no reconhecimento e domínio de padrões de um jogo [34].

Procura-se entender o que dá prazer ao jogar, o nível de desafio almejado, a motivação de estar jogando, 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 [43]. A compreensão do relacionamento do jogador com o jogo é importante para alcançar uma melhor experiência do usuário, podendo ser classificado de acordo com as motivações para preferir determinados jogos ou possuir comportamentos específicos ao jogar.

2.2.1 Classificação de jogadores

Os jogadores podem ser divididos em grandes categorias por fatores comportamentais, demográficos ou psicológicos [16]. Dessa forma, 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.

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 Salen e Zimmerman [35] os jogadores podem ser divididos em tipo de acordo com a aderência às regras: casual, dedicado, antiesportivo, trapaceiro e desmancha-prazeres. Outra categorização popular pela comunidade de desenvolvedores de jogos é feita entre jogadores pelo nível de experiência [16]:

- Jogadores casuais jogam com menor frequência, preferindo jogos rápidos que não exigem tanta dedicação e proporcionam uma distração rápida. Relacionam-se aos jogadores casuais descritos por [35], jogando conforme foi designado, seguindo as regras e respeitando a autoridade.
- Jogadores experientes empregam bastante tempo em jogos e inclinam-se a escolher jogos mais complexos, que possuem narrativas ou mecânicas mais elaboradas. Relacionam-se aos jogadores dedicados de [35], procurando dominar e se aperfeiçoar no jogo, geralmente encontrando e explorando estratégias incomuns para ganhar.

Os jogadores casuais não abordam o mesmo conjunto de habilidade dos jogadores experientes, possuindo níveis diferentes de exploração do jogo e de paciência com falhas [44]. 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.

2.2.2 Emoção do jogador

Comportamentos humanos complexos emergem de interações dinâmicas entre múltiplos processos e redes cerebrais, interligando emoção e cognição [45]. Embora existam muitas perspectivas diferentes sobre emoção, a estrutura cognitiva da teoria da avaliação é a que mais se adéqua à modelagem computacional [46]. Essa teoria define emoções como respostas adaptativas que são elicitadas em base a como o agente avalia uma situação, associando diretamente aos objetivos motivacionais do agente e ocorrendo em resposta a estímulos externos e internos gerados por pensamentos. Com isso, agentes interpretam o ambiente continuamente e os estados emocionais variam a partir da mudança das informações.

As emoções advêm de avaliações cognitivas e organizam respostas de comportamento adaptativas [46]. Logo, correspondem a estados psicológicos compostos de múltiplos processos inter-relacionados, como avaliações cognitivas, respostas fisiológicas, tendências de ação comportamental e a experiência fenomenológica dos sentimentos.

De acordo com Lang [47], as emoções são nossa força motriz, definindo se uma ação será executada. Assim, pode-se considerar que as emoções são provisões e não ações isoladas, classificando-as em dois componentes inter-relacionados e dependentes: excitação e valência. Além disso, são expressas fisicamente em sinais como sudorese, fluxo sanguíneo cutâneo (corar ou empalidecer), piloereção (elevação involuntária de pelos no corpo), alteração da frequência cardíaca e condutância da pele, sendo alguns casos detectáveis e avaliáveis pela análise empírica de medições de dispositivos e sensores de *biofeedback*.

Essa classificação é relacionada ao modelo Circumplexo apresentado por Russell [2]. As emoções são descritas em duas dimensões nesse modelo, a valência afetiva (eixo x) e o nível de excitação (eixo y), ilustrado na Figura 2.2. Nessa visão dimensional [47], a valência se refere à relativa vivacidade da experiência emocional (desagradável ou prazerosa), identificando a positividade ou negatividade inerente de uma emoção. Por outro lado, a excitação se refere à intensidade (baixa ou alta) da experiência emocional (variando de calmo à excitado).

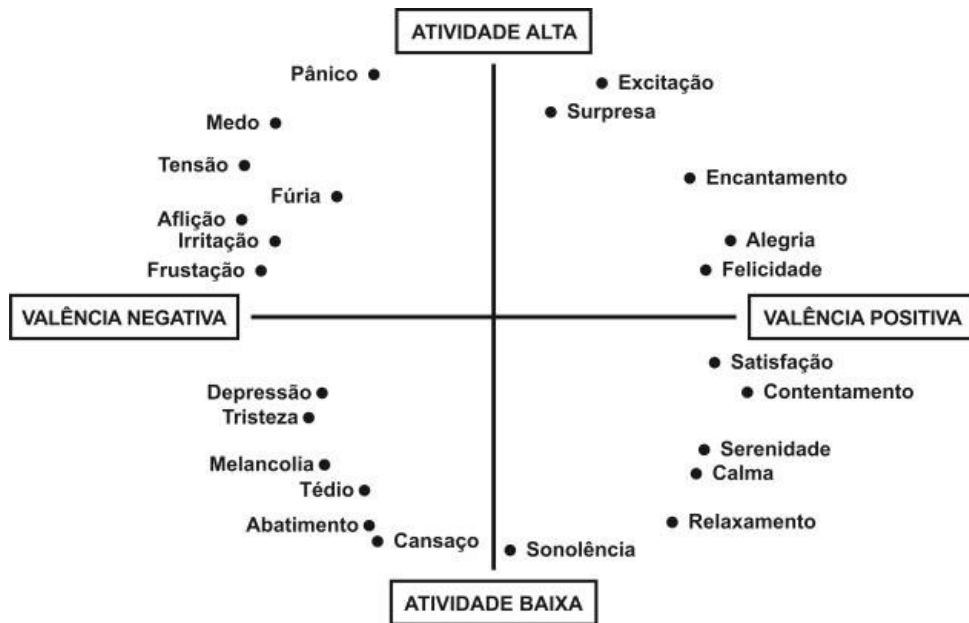


Figura 2.2: Modelo Circumplexo de Afeto com as emoções geradas pela combinação de excitação e valência (adaptado de [2]).

Emoções são compostas por uma mistura de ativação fisiológica, comportamentos expressivos, pensamentos e sentimentos [48]. As manifestações fisiológicas das emoções são comandadas pelo sistema nervoso autônomo, o qual é responsável pela resposta de fuga ou luta e por outras funções corporais involuntárias. O sistema nervoso autônomo é composto por duas divisões: simpática e parassimpática, respectivamente responsáveis pela indução de glândulas e liberação de hormônios, e para que o corpo volte a um estado homeostático, de equilíbrio no funcionamento habitual.

As emoções são uma parte vital da experiência do jogo e possibilitam análises para manter o jogador em uma zona de valência determinada, considerando que diferentes níveis de dificuldade podem induzir estados afetivos específicos e influenciarem no engajamento [26]. Ademais, emoções específicas ativam diferentes atividades cerebrais e resultam em distintas respostas corporais, permitindo vincular alterações fisiológicas a estados afetivos, mesmo que não seja possível perceber visualmente. Esse vínculo pode ser realizado ao analisar o eixo do nível de excitação, detectando e interpretando medidas fisiológicas.

2.2.3 Afetividade

A afetividade é o conjunto de fenômenos psíquicos experimentados e vivenciados na forma de emoções e sentimentos e está relacionada ao *biofeedback*, mecanismo de obtenção de variáveis do corpo humano, respondendo ao estímulo em tempo real [49]. O *biofeedback* pode ser separado em duas categorias abrangentes: biomecânicos e fisiológicos. A primeira está relacionada a movimentos e habilidades motoras, enquanto a segunda se refere a variáveis do corpo humano como temperatura, batimento cardíaco e condutividade da pele.

Esse mecanismo possui um papel importante no campo de estudo de jogos, apoiando a aquisição de dados do usuário. Considerando que há uma variedade de sensores disponíveis, deve-se analisar a adequação desses para contextos específicos, por exemplo para jogos de ação e terror em comparação a jogos de quebra-cabeça pela carga de estresse relativa [50].

Atividade Eletrodérmica

A Atividade Eletrodérmica (EDA) ou Resposta Galvânica da Pele (GSR) está geralmente relacionada a alguma reação do sistema nervoso simpático, indicando a mudança do nível de entusiasmo [49]. Isso se deve à secreção das glândulas sudoríparas, um processo que permite que o corpo regule sua temperatura. Todavia, o sensor pode apresentar falhas por diferentes padrões gerados para diferentes peles humanas.

A resistência da pele muda quando há excitação ou relaxamento pelo estado emocional ser parcialmente traduzido na produção de suor ou na inibição das glândulas em mãos, palmas e pés. A condutância elétrica aumenta de modo mensurável, possibilitando a tradução dessas mudanças em valores numéricos por meio do monitoramento da atividade eletrodermal.

Caso ocorra uma ativação emocional, um aumento da carga cognitiva ou um maior esforço físico, o cérebro envia sinais para a pele aumentar o nível de suor, ainda que não seja sentido. Esse sensor pode ser utilizado para o mapeamento afetivo e o *biofeedback* de estresse/relaxamento.

2.2.4 A experiência do jogador

Uma experiência otimizada possibilita a sensação de prazer durante sua ocorrência, quando o corpo ou a mente é aplicado ao seu limite em um esforço voluntário para realizar algo difícil ou satisfatório [51]. As experiências são representadas na mente como informação e estados ótimos são resultado da ordem na consciência. Ou seja, isso ocorre quando há

um foco em objetivos realistas com as habilidades individuais combinado com as oportunidades de ação e, assim, o indivíduo pode se concentrar na tarefa em questão.

As abordagens utilizadas para maximizar experiência e consequente satisfação do jogador podem ser classificadas em duas categorias [52]: implícitas, consistindo na alteração de parâmetros periféricos à satisfação como variáveis do jogo e explícitas, que maximizam o valor de uma função representativa da satisfação do jogador. Nesse contexto, o mecanismo de Ajuste Dinâmico de Dificuldade utilizada abordagem implícita e as adaptações podem ser realizadas a partir de dados provenientes do jogo ou obtidos com sensores fisiológicos.

Perfil do jogador

Para melhorar a experiência do jogador, além da medição da dificuldade do jogo em correspondência a sua habilidade, os jogadores podem ser categorizados em perfis diferentes. Isso possibilita o aprimoramento da lógica de adaptação quando comparada às adaptações baseadas exclusivamente no desempenho. Nesse sentido, uma das taxonomias mais conhecidas é a proposta por Bartle [53], pensada originalmente para jogos Multiuser Dungeons (MUD). Os jogadores foram classificados por suas características psicológicas entre:

1. Realizadores (querem pontos e conquistas);
2. Exploradores (querem ver novas coisas e descobrir segredos);
3. Assassinos (querem pontos por competição e derrotar inimigos); e
4. Socializadores (querem diversão por interação com outros jogadores).

Outros modelos foram sendo propostos durante os anos, com similaridades identificáveis e generalização para distintos gêneros de jogo. Assim, há um modelo de reconhecimento de perfis de jogador chamado ADOPTA [54]. Este modelo é baseado na teoria de aprendizagem experiencial de Kolb [55] e associa estilos de aprendizagem [56] a quatro perfis de jogador [54]:

1. Competidores (confiam na intuição e assumem riscos);
2. Sonhadores (observam ao invés de controlar a jogabilidade);
3. Lógicos (preferem análises e abordagens baseadas em padrões); e
4. Estrategistas (gostam de resolver problemas complexos de maneira eficaz).

O reconhecimento de perfil de jogador pode se basear também no perfil de risco do jogador. Ao considerar a recompensa do jogador em termos de satisfação geral, uma divisão típica de estilos de gerenciamento seria busca de risco, risco neutro e avesso ao risco [11]. Esses três perfis de risco diferentes relacionam utilidade e recompensa. Os caçadores de risco ficam satisfeitos apenas com grandes recompensas e riscos maiores, respondendo mais rápido, mas com maior probabilidade de fracasso. Enquanto isso, jogadores avessos ao risco preferem decisões de menor risco e recompensa, esperando mais por mais evidências e, assim, aumentando a probabilidade de sucesso.

Um modelo mais recente, proposto por Marczewski [3][57], associa sistemas a personalidades de usuário, sendo chamado de Hexad. Esse modelo é baseado em estudos sobre motivação humana, tipos de jogadores e experiência prática em *design*, possuindo associação significativa com o modelo de fatores globais de personalidade Big Five [58].

Nesse contexto, seis tipos de perfis foram propostos, conforme visto na Figura 2.3, diferindo no grau em que são motivados por fatores intrínsecos, como auto-realização, ou extrínsecos, como recompensas:

- Realizador (motivado por maestria);
- Socializador (motivado por conexões);
- Filantropo (motivado por propósito);
- Espírito Livre (motivado por autonomia);
- Jogador (motivado por recompensa); e
- Revolucionário (motivado por mudança).



Figura 2.3: Tipos de usuário Hexad [3].

Os quatro primeiros tipos são intrinsecamente motivados, enquanto Jogadores necessitam receber recompensas do sistema e Revolucionários querem perturbar o funcionamento do sistema e precisam de algo que os mantenha interessados. Além disso, observa-se que usuários não podem ser divididos categoricamente nos tipos mencionados, possuindo características múltiplas e em graus variados. O objetivo de categorizar jogadores em perfis é a disponibilização de um método simples sobre os possíveis usuários que usarão um sistema.

2.3 Fluxo

Conforme apresentado por Mihaly Csikszentmihalyi [51], o prazer (*pleasure*) pode ser definido como o sentimento de contentamento alcançado quando uma informação na consciência indica que as expectativas estabelecidas por programas biológicos e por condicionamentos sociais foram atingidas. Por sua vez, o prazer (*enjoyment*) pode ser visto como o resultado da pessoa atingir uma expectativa, ir além do que estava programada a fazer e alcançar algo inesperado. Ou seja, é percebido como o sentimento de novidade ou de realização, possuindo 8 componentes principais:

1. Tarefas com uma possibilidade razoável 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.

Os termos supracitados foram analisados para definir o conceito de fluxo [51], o qual pode oferecer uma explicação teórica para o prazer (*enjoyment*). Durante o estado de fluxo, a atenção é investida livremente para atingir os objetivos de uma pessoa, pois não há desordem para fortalecer ou uma ameaça para se defender. Assim, o fluxo ocorre quando há um balanço entre as habilidades do indivíduo e a dificuldade da tarefa apresentada.

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 à integração porque pensamentos, intenções, sentimentos e sentidos estão focados no mesmo objetivo, facilitando as conexões e a segurança.

A teoria do fluxo indica ser possível para qualquer pessoa experimentar o fluxo em qualquer atividade [59]. Todavia, um considerável investimento de tempo e esforço na atividade são requeridos, além de atender os seguintes pré-requisitos: as habilidades devem ser compatíveis com os desafios propostos em uma proporção correta e o indivíduo precisa ter uma personalidade autotélica, sendo capaz de reconhecer e aproveitar oportunidades de fluxo.

Por outro lado, considerando o isomorfismo do fluxo [51], deve haver certos tipos de atividade universalmente acessíveis que antecipam e capacitam estados cognitivos universais e inerentes à experiência do fluxo [59]. Com isso, algumas atividades facilitam o fluxo para qualquer indivíduo, independente de sua habilidade, como é o caso dos jogos. Observa-se que as consequências de um usuário entrar no estado de fluxo são: um aumento do autocontrole, a concentração na atividade em questão, uma clara visão dos objetivos, a perda de autoconsciência, uma sensação alterada do tempo, um aumento na motivação pessoal e uma fusão de ação e percepção.

Os aspectos do fluxo são mantidos para mídias nos formatos tradicional e interativo, como televisão e jogos eletrônicos [60]. Dessa forma, um indivíduo experimenta entretenimento de uma mensagem de mídia dependendo da capacidade de atingir o estado de fluxo através desse meio. Por conseguinte, o design de mídias pró-sociais é facilitado pela compreensão dos mecanismos subjacentes ao prazer e o engajamento é facilitado ao associá-lo com o estado de fluxo, prolongando o uso do material.

2.3.1 Fluxo em jogos

Um fator comum a jogos de sucesso é trazer 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 exterior. Um indivíduo pode chegar ao ponto de não perceber o tempo passando e os acontecimentos ao seu redor por estar com a atenção quase totalmente voltada ao jogo, o que resulta na sensação de que é o próprio personagem que controla o jogo. De acordo com Cowley, ao atingir esse estado, o jogador está imerso [59].

A imersão pode ser vista como um fator crítico para o aproveitamento do jogo e uma boa experiência do jogador. Chen aponta que esse estado é ocasionado por uma atividade com desafio suficiente para que não se perca o interesse por tédio e que não supere suas habilidade e se sinta frustrado. Nesse contexto, os jogos podem ser analisados por dois componentes essenciais [4]:

- Conteúdo do jogo: uma experiência específica, a qual o jogo é projetado para transmitir.
- Sistema do jogo: um software interativo que comunica o conteúdo do jogo para os jogadores por intermédio de visuais, áudio e interações.

Quando tratado como um sistema, o fluxo explica por que as pessoas preferem certos jogos [4]. No caso de um jogo alcançar todos os elementos principais do fluxo, qualquer conteúdo poderá ser recompensador e qualquer premissa poderá ser engajadora [61].

Cada jogador possui uma curva de experiência obtida ao jogar, como visto na Figura 2.4. Caso a experiência se afaste da zona de fluxo, a entropia psíquica negativa (ansiedade, tédio) irá interromper o estado de fluxo. Além disso, pessoas diferentes têm diferentes habilidades e zonas de fluxo. Ou seja, um jogo bem projetado pode manter os jogadores casuais no fluxo, porém não ser tão eficaz para jogadores experientes. Quanto maior a habilidade do usuário, maior deve ser o desafio para que o usuário se mantenha no estado de fluxo.

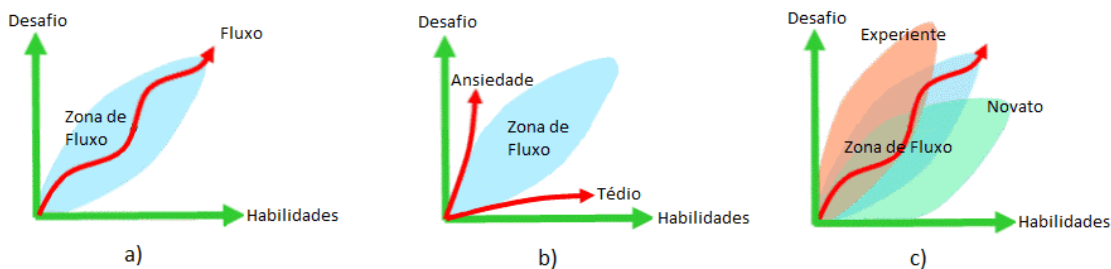


Figura 2.4: a) Experiência do estado de fluxo em um jogador; b) Jogador ao encontrar entropias psíquicas; c) Diferentes jogadores e zonas de fluxo (adaptado de [4]).

Para criar um jogo para públicos mais abrangentes, a experiência no jogo não pode ser linear e estática e precisa oferecer uma ampla cobertura de experiências para se encaixar nas zonas de fluxo de diferentes jogadores. Um ajuste de desafio pode ser utilizado objetivando esse equilíbrio, mantendo o desafio proporcional à habilidade.

2.4 Dificuldade em jogos

Os jogos eletrônicos têm como função generalista a diversão, de acordo com Huizinga [33]. Entretanto, não há um caminho definido para alcançar isso, dependendo de diversos aspectos como a dificuldade. Assim, o dimensionamento da dificuldade é fundamental ao projeto do jogo, sendo definido pela configuração de obstáculos para serem superados voluntariamente [62]. Portanto, um bom *design* de jogo deve ter uma curva de aprendizado e dificuldade coerentes [63].

O sentimento subjetivo da dificuldade é complexo e não está ligado somente ao resultado final do desafio [62]. No caso, a dificuldade é derivada da relação entre habilidade e desafio, sendo que a dificuldade varia de acordo com a habilidade necessária para resolver um desafio [20]. Assim, um jogo é entediante quando fácil demais e frustrante quando muito difícil.

Em vista da dificuldade ser subjetiva, variando de acordo com a habilidade do jogador, existem maneiras de modular o grau de dificuldade oferecido e influenciar o conteúdo apresentado. Habitualmente existe a separação interna ao jogo por níveis de dificuldade, como fácil, médio e difícil. Todavia, há um crescente campo para a adaptatividade em jogos, como o Ajuste Dinâmico de Dificuldade.

Existem casos de jogos que não aplicam diferentes dificuldades, como *Space War!*, *Pong* e *GunFight*. Isso se deve a simplicidade atrelada a esses jogos, com desafios únicos e habilidades constantes [43]. Outros jogos fazem uma escala do desafio, aumentando gradativamente a cada nível (*Mario*, *Mega Man*) ou feita com o nível do personagem (*Elder Scrolls*). Ademais, alguns jogos adaptam o desafio à habilidade do personagem (*Mario Kart*), possuem opções acerca do foco do jogador na narrativa ou na ação (*Mass Effect*), ou sugerem uma mudança de nível no caso de muitas derrotas consecutivas (*God of War*). Nesse cenário, alguns padrões informais de representação da dificuldade são:

- Surpresa não é o mesmo que dificuldade, ou seja, a existência de eventos sem um padrão definido não contribui para dificultar um desafio se o fator surpresa correspondente não pode ser dominado, então o desafio não tem uma motivação de fundo.
- Dificuldade implica em falha possível e provável. No caso, falha provoca a perda de algo e, quando a perda aumenta, o mesmo acontece com a dificuldade geral percebida. Esse fator é visto quando o progresso no jogo é salvo e quando há uma quantidade limitada de vidas.
- Reduzir as janelas de tempo de uma ação ou um conjunto de ações aumenta a dificuldade.
- Uma sequência contínua de ações tem maior dificuldade do que as mesmas ações realizadas de forma independente, com separações estruturais entre cada uma dessas ações.
- 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 específico.

Com isso em vista, Juul [32] expande formalmente os princípios de representação da dificuldade ao explorar os padrões de *design* populares para representar a dificuldade no jogo. Nesse caso, analisa a implementação da falha, a qual é descrita como sendo representada das seguintes maneiras:

- Punição de recuo, forçando o usuário a reproduzir uma certa parte do nível.
- Punição de energia, que serve como um aviso de uma penalidade mais forte, mesmo que isso não implique penalidade direta no momento.
- Punição de vida, forçando um contratempo na sequência de jogo contínua a um certo ponto de verificação.
- Punição de terminação de jogo, significando que o jogador perde completamente o desafio e tem que reiniciar o jogo desde o começo.

2.4.1 Adaptatividade

Os sistemas adaptativos são utilizados para fornecer ao jogador uma experiência personalizada, ajustando o jogo de acordo com objetivos identificáveis, mensuráveis e influenciáveis, conforme definido por Lopes [15]. Geralmente são definidos por um fator dinâmico, como a habilidade do jogador, e realiza ajustes baseado nesses objetivos [20].

Por intermédio desses sistemas, Novak apontou o objetivo de alcançar o equilíbrio do jogo para cada jogador, sendo este um estado no qual o jogador percebe que o jogo é consistente, justo e divertido [16]. Além disso, é dividido em equilíbrio estático e dinâmico: relativo às regras e mecânicas do jogo e à interação do jogador com o jogo, respectivamente. Consoante com o esperado de uma experiência que possibilite entrar no estado de fluxo, para o jogo estar equilibrado, a jogabilidade deve fornecer:

1. Desafios compatíveis com a habilidade do jogador;
2. Uma experiência justa;
3. Ausência de estagnação;
4. Ausência de decisões banais; e
5. Níveis de dificuldade, podendo ser escolhido ou ajustado.

Todavia, a determinação e quantificação da expectativa do jogador é necessária para adaptar conforme a motivação do jogador e a realização do processamento de sinais [15]. Para auxiliar nisso, jogos podem ser separados em 3 categorias conforme a adaptatividade da dificuldade associada [20]:

- Não-adaptativos: não oferecem um mecanismo adaptativo de dificuldade, incluindo um único nível de dificuldade ou permitindo que o jogador escolha qual nível o satisfaz. Exemplos: *Pong* e *Mass Effect*.
- Parcialmente adaptativos: possuem uma forma automática de detectar se a habilidade do jogador é superior à dificuldade atual, mas não realizam a mudança automaticamente, ou possuem um mecanismo que escala a dificuldade somente em uma direção, seja aumentando ou diminuindo para se manter compatível com o jogador. Exemplos: *Elder Scrolls*, *God of War* e *Mega Man Zero*.
- Adaptativos: possuem um mecanismo dinâmico de ajuste de dificuldade, aumentando ou diminuindo automaticamente a dificuldade de forma a balancear o jogo em relação à habilidade do jogador. Exemplos: *Mario Kart*, *Left 4 Dead* e *Max Payne*.

A variação de dificuldade proporcionada pelos jogos adaptativos relaciona-se à evolução do jogador (habilidades e conhecimentos) e/ou do seu avatar (mecânicas para vencer desafios) [20]. Ademais, observa-se que é natural que jogos gradualmente aumentem a dificuldade de modo proporcional à progressão do jogo, porém isso não se caracteriza como adaptatividade. No caso, a adaptação devido a aprendizagem natural se distingue de ajustes específicos ao jogador.

Um modelo de adaptatividade para jogos afetivos [64] pode ser centrado no jogador e baseado nos conceitos da teoria clássica do controle, a qual lida com o comportamento de sistemas dinâmicos como entradas e as modificações dessas pelo *feedback*. Esse modelo mostra como um parâmetro do jogo é controlado pelo nível de excitação do jogador estimado pela medição de uma resposta fisiológica (EDA). Como visto na Figura 2.5, o modelo tenta manter o valor do dado fisiológico medido no tempo ($r(t)$) ao usar um laço afetivo, assim a excitação atual do jogador permanece em um nível ótimo (r_0). Por outro lado, é mensurado um valor de estado emocional ($\varepsilon(t)$) e o nível de estímulo do jogo no tempo é usado como estímulo emocional para o jogador ($s(t)$).

O modelo é genérico para jogos afetivos adaptativos e utiliza um controle de *feedback* negativo, mantendo a excitação do jogador perto do nível ideal para melhorar sua atenção e possibilitar um desempenho superior durante o jogo. Os controles negativos reduzem a diferença entre a medida psicofisiológica em tempo real e o nível desejado do estado emocional do jogador. Por conseguinte, cria-se estabilidade comportamental durante o jogo e o jogador é mantido dentro de uma zona segura e estável de eficácia no desempenho [65].

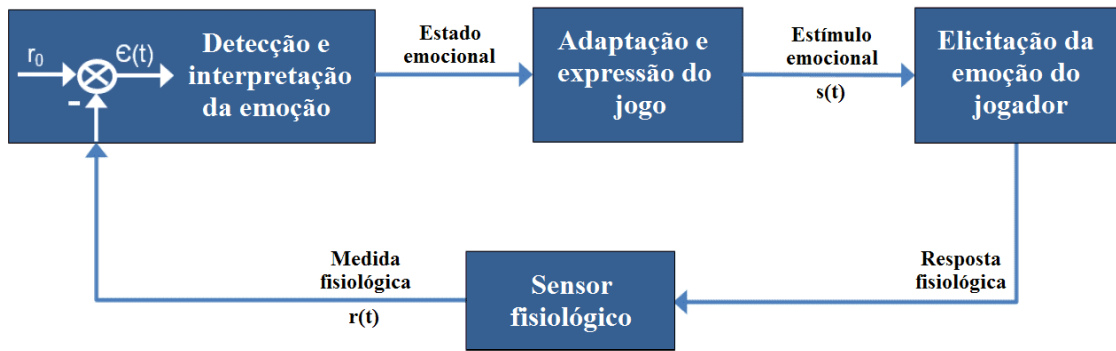


Figura 2.5: Diagrama de blocos de um jogo afetivo adaptativo (adaptado de [5]).

2.4.2 Ajuste Dinâmico de Dificuldade

O Ajuste Dinâmico de Dificuldade (ADD) em jogos eletrônicos é uma técnica emergente que procura adaptar o nível de dificuldade de um jogo enquanto esse é jogado para poder se adequar às habilidades de jogadores específicos [19]. Serve como um contraponto a níveis de dificuldade pré-estabelecidos, podendo alterar de acordo com o gênero e os objetivos do *designer*.

Os jogos com dificuldade não-adaptativa podem não conseguir atender um público de variados comportamentos e habilidades ao causar tédio ou estresse não planejados nos jogadores por não avaliar a correspondência entre o grau do desafio e o perfil do jogador. Ainda assim, como os custos de implementação e a complexidade do ADD são elevados, há uma dificuldade em evoluir esse mecanismo para englobar todos os jogos [18]. Nesse sentido, existem três requisitos para a adaptatividade dinâmica em jogos [19]:

1. Identificar e se adaptar ao nível do jogador o mais rápido possível;
2. Perceber e registrar as mudanças de desempenho do jogador; e
3. Manter o comportamento do jogo, durante a adaptação, verossímil e discreto, de modo que o jogador não perceba o sistema.

Pode haver uma perda de imersão no jogo ou abuso do jogador por meio da manipulação das mudanças, caso o uso do ADD se torne perceptível. Por outro lado, se o jogo demora a se adaptar, o ADD pode não corresponder de modo eficaz às habilidades do jogador e assim não ter o efeito desejado. Esse tipo de problemática interfere negativamente na experiência ao ocasionar a quebra de verossimilhança, a falta das sensações de domínio sobre o desafio e reações negativas para ajustes executados [17].

Logo, o Ajuste Dinâmico de Dificuldade modifica, em tempo real, fatores e variáveis do jogo para adequar o desafio ao jogador. Para isso, tipicamente são implementados 4 módulos interligados [4]:

- Jogador: cria os dados brutos ao jogar;
- Sistema de Monitoramento: grava métricas brutas da desempenho do jogador, com dados de sensores especializados também podendo ser coletados e associados;
- Sistema de Análise: computa os dados brutos recebidos para determinar quais componentes do jogo devem ser ajustados; e
- Sistema de Controle: ajusta os componentes de jogo conforme especificado pelo Sistema de Análise.

Desse modo, o ADD deve funcionar como uma malha de controle que monitora o jogador e as variáveis do jogo e age em conformidade. Como ilustrado na Figura 2.6, pode ser representado por um ciclo que relaciona os módulos mencionados com os dados medidos, suas análises e as adaptações correspondentes.



Figura 2.6: Ciclo de ADD orientado pelo sistema (adaptado de [4]).

Os métodos de adaptação podem ser divididos em: ajuste de parâmetros, métodos comportamentais e técnicas de aprendizado [20]. Observa-se que, para direcionar a adaptatividade, a maioria depende de um certo nível de modelagem do jogador e que o ajuste de parâmetros pode ser classificado em 3 grupos principais [43]:

- Atributos do personagem principal (avatar), como vida, distância de salto e dano infligido no inimigo.

- Atributos de Personagem Não Jogável (NPC), sendo atributos fundamentais, como vida e impacto do ataque, ou processos de decisão, como algoritmos de procura de caminho (*path-finding*).
- Variáveis de mundo e de nível, como distância entre plataformas e posicionamento de itens colecionáveis. Dessa forma, o jogo pode ser projetado para o mundo ser alterado sutilmente de maneira a ajudar ou dificultar as ações do jogador.

2.4.3 Dificuldade em jogos de plataforma

A adaptação de dificuldade é particular ao gênero de cada jogo, resultando em implementações específicas. Logo, diferentes componentes podem ser adaptados pelo Sistema de Controle em distintos gêneros [5], como a habilidade de carros oponentes em jogos de corrida, a velocidade do avatar ou a densidade de inimigos para jogos *First Person Shooter* (FPS). Além disso, essa particularidade também pode ser vista no Sistema de Análise, com o ADD se baseando em dados como a economia de itens em jogos *First Person Shooter* (FPS) ou a porcentagem de itens colecionáveis coletados em jogos de plataforma.

Ou seja, o ADD é feito a partir de componentes intrínsecos ao gênero adaptado, assim não podendo ser replicado entre eles. Especificamente para analisar a dificuldade em jogos de plataforma, é importante especificar a construção de um jogo desse gênero, considerando os seus componentes e o modo como se interligam para formar um nível.

Observa-se que os componentes são as unidades básicas das quais os jogos de plataforma são construídos, assim como vinhas, plataformas, colinas e picos [66]. Pode ser tanto um obstáculo quanto um local de descanso e possuir propriedades físicas, como potencial de lesão e ser escorregadio. Assim, os componentes de um jogo de plataforma podem ser categorizados pelo seu propósito no nível do jogo, assim como exemplificado pela Figura 2.7 baseado nos padrões de *design* do jogo *Super Mario Bros.*. Observa-se também que a categorização dos componentes pode ser feita enfatizando a estrutura em vez da representação visual [7]:

- Plataformas: qualquer objeto que o jogador executa ou pula para atravessar o nível, como superfícies planas, laços ou o topo de caixas de item.
- Obstáculos: qualquer objeto que seja capaz de causar dano ao avatar. Observa-se que um buraco entre duas plataformas também é considerado obstáculo, mesmo que não sejam explicitamente objeto no nível.
- Ajudas de movimento: qualquer objeto que ajuda o jogador a atravessar o nível, como molas, trampolins móveis ou cordas.

- Itens colecionáveis: qualquer objeto que fornece uma recompensa ao jogador, como uma moeda, poder ou recompensa pontual.
- Gatilhos: qualquer objeto que de alguma forma altera o estado do nível. Exemplos de gatilhos incluem chaves que transformam blocos em moedas e objetos que alteram o comportamento do avatar.

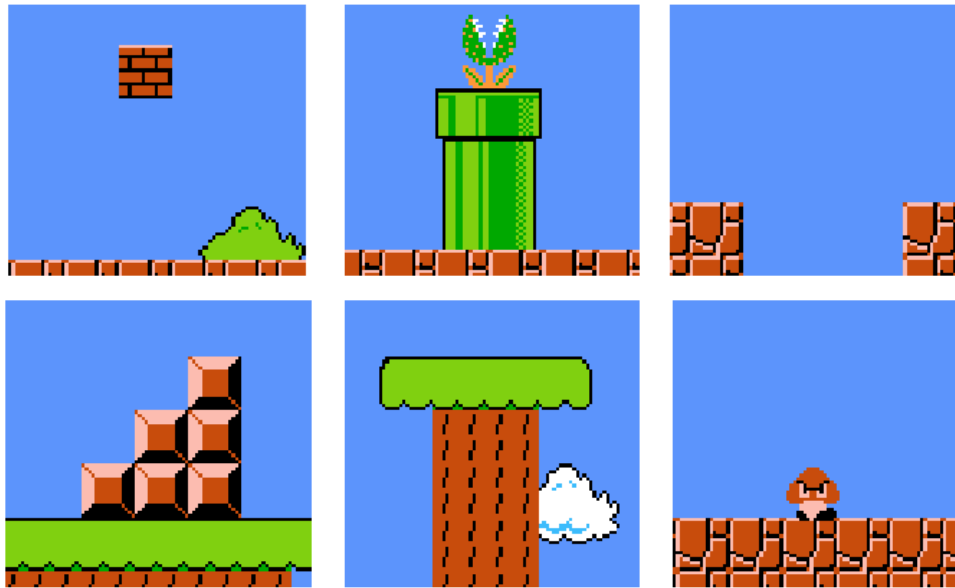


Figura 2.7: Elementos de design para o jogo *Super Mario Bros.*, em sentido horário: bloco, cano, buraco, inimigo, plataforma e escadaria [6].

As células são os blocos de construção do design de um nível não linear, feitas a partir do encapsulamento de padrões lineares [66]. Ou seja, são a representação estrutural da relação dos componentes como uma hierarquia [7], apresentado na Figura 2.8, contendo os componentes na parte inferior e o nível completo no topo. Dessa forma, há um enfoque em dividir o nível em áreas de desafio:

1. A maior subdivisão de um nível é chamada de célula, que é uma seção de jogabilidade linear. As células estão ligadas por portais.
2. As células são compostas de grupos rítmicos, os quais são conjuntos não sobrepostos dos componentes do nível. Esses grupos rítmicos são geralmente pequenos, encapsulando seções desafiadoras do jogo. Intervalos em grupos de ritmo são locais seguros para o jogador descansar antes de continuar o nível.

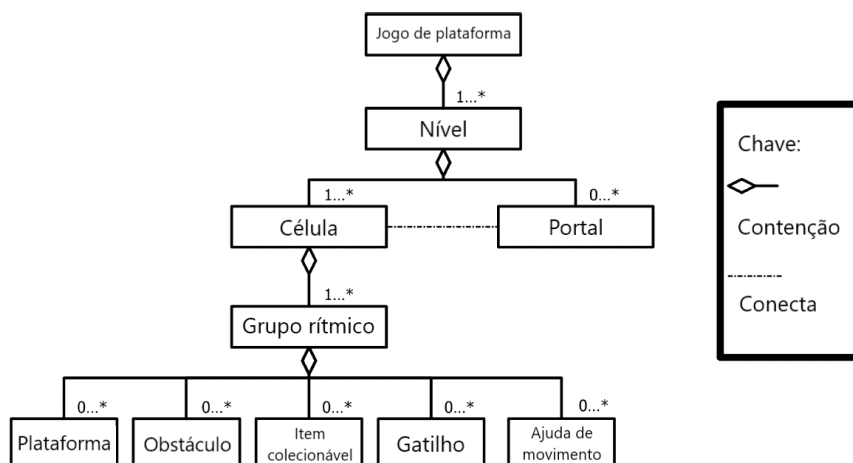


Figura 2.8: Ilustração do modelo conceitual de estrutura de níveis (adaptado de [7]).

2.5 Aprendizado de Máquina

A Inteligência Artificial (IA) consiste em uma tecnologia complexa que possibilita a interação com o meio ambiente e simula a inteligência humana. A IA transforma o conhecimento sobre a relação entre o ser humano e a tecnologia, sendo considerada um dos tópicos mais importantes para o desenvolvimento econômico e social [67].

Os sistemas de IA se associam a novas formas de aprendizagem com a complexidade das funções da IA tornando seu comportamento não determinístico [68]. Nota-se que os recursos de IA para resolver problemas são aplicados de forma eficaz em vários campos de aplicação, como robótica, indústrias e jogos. Correspondem a tecnologias capazes de interagir por meio de coleta de informações, interpretação, reconhecimento de padrões, indução de regras e previsão de eventos. Nesse sentido, realizam tarefas em diversos ambientes e podem gerar resultados, fornecer instruções e avaliar os resultados de sua ação, melhorando seus sistemas de decisão [69].

O Aprendizado de Máquina é um componente comumente usado de IA, permitindo que um computador ajuste seu comportamento com base nos dados expostos. Nesse caso, com base em um objetivo, uma IA devidamente treinada pode tomar decisões com precisão e ajustar comportamentos de acordo com o processamento de um dado existente [70]. Um algoritmo de Aprendizado de Máquina possibilita identificar padrões de propriedades funcionais, aprender e melhorar sem reprogramação, podendo ser dividido em:

- Aprendizado supervisionado, no qual o modelo aprende a partir de resultados pré-definidos, sendo apresentadas entradas e saídas desejadas e objetivando aprender uma regra geral de mapeamento das entradas para as saídas.
- Aprendizado não supervisionado, no qual o modelo não possui resultado pré-definidos de referência, encontrando sozinho uma estrutura nas entradas fornecidas.

- Aprendizado por reforço, no qual o modelo interage com um ambiente dinâmico, desempenhando um objetivo determinado e recebendo um parecer quanto a premiações e punições enquanto percorre o espaço do problema.

No campo dos jogos eletrônicos, o Aprendizado de Máquina é usado para proporcionar uma experiência interessante e interativa para o jogador [71]. Pode ser utilizado para a adaptação do jogo, buscando equilibrar o nível de desafio de diferentes jogadores de acordo com seu nível de habilidade, objetivo principal do Ajuste Dinâmico de Dificuldade [72]. Além disso, pode influenciar a motivação e o prazer geral dos jogadores enquanto diferentes objetivos de pesquisa são definidos, como a criação de um NPC mais complexo e a compreensão dos jogadores e seus padrões de interação.

2.5.1 Rede Neural Artificial

A Rede Neural Artificial é um modelo computacional baseado na estrutura e no funcionamento de redes neurais biológicas, imitando como células nervosas funcionam no cérebro humano e sendo capazes de realizar o aprendizado de máquina [73]. Nesse cenário, algoritmos de aprendizado são utilizados de forma a fazer ajustes independentemente quando recebem novas entradas. Assim, são vistas como ferramentas de modelagem de dados estatísticos não-lineares, nas quais as relações entre entradas e saídas são modeladas e há o reconhecimento de padrões.

Uma RNA possui três ou mais camadas interconectadas, sendo a primeira de neurônios de entrada que enviam dados para as camadas mais profundas, as quais enviarão a saída para a última camada. Todas as camadas internas são ocultas e formadas por neurônios que se adaptam a informação recebida de cada camada, recebendo uma entrada, multiplicando-a por um peso, processando com uma função de ativação e passando a saída para camada seguinte. Nessas camadas, busca-se aprender sobre a informação coletada dando pesos de acordo com o sistema interno da RNA.

Pelo aprendizado supervisionado, a rede recebe dados de entrada com a resposta desejada de saída, com os dados de entrada geralmente propagados para frente na rede até que a ativação atinja um dos neurônios de saída [73]. Com o uso de retropropagação, erros podem ser considerados e a rede pode ser treinada. Nesse contexto, a informação retorna quando a saída é rotulada como errada durante a fase de treinamento supervisionado, atualizando os pesos proporcionalmente ao quanto foram responsáveis pelo erro e podendo minimizar a chance de erros e resultados indesejados.

Uma das vantagens de uma RNA é a capacidade de aprender a partir da observação de um conjunto de dados, sendo um campo que permite encontrar estruturas de dados e algoritmos para aprendizado de regressão e classificar dados [74]. O uso mais comum

de RNA é da modelagem de agentes que imitem comportamentos de jogadores [75], mas também pode ser utilizada em outros contextos, como classificar jogadores conforme suas características.

Capítulo 3

Trabalhos Correlatos

Neste capítulo serão expostos trabalhos relacionados cujas investigações e resultados foram úteis no âmbito deste trabalho. Cada seção destaca um assunto relevante e os pontos principais das pesquisas estudadas, como a avaliação da dificuldade e a aplicação do Ajuste Dinâmico de Dificuldade em jogos de gêneros variados e, especificamente, em jogos de plataforma.

3.1 Avaliação da dificuldade

A dificuldade é um tema de estudo recorrente nos diversos gêneros de jogos eletrônicos. Existem inúmeros métodos para avaliação da dificuldade, variando conforme as especificidades de um gênero ou a metodologia proposta por cada estudo.

3.1.1 *Design* do desafio

Estratégias de *design* definem o equilíbrio do jogo e a conseqüente experiência do jogador. A manipulação de certas decisões de projeto e suas influências na dificuldade foram investigadas por Wehbe et al [76]. Nesse estudo, foram analisados a velocidade de rolagem da câmera, o tamanho do alvo (plataforma), a complexidade do salto e a perspectiva (direção de rolagem da câmera) em um jogo de plataforma com movimento automático da câmera.

Foram testados nove segmentos com diferentes escolhas de *design* com 16 participantes. Cada jogador foi avaliado em uma escala de dificuldade e dados de execução com sucesso dos saltos do jogador foram salvos. Os resultados para velocidade de rolagem e tamanho do alvo indicam que os erros aumentam conforme a velocidade aumenta e o tamanho da plataforma diminui. Todavia, a complexidade da tarefa de salto não possui, necessariamente, uma associação direta com o aumento de dificuldade. Ou seja, embora as tarefas de salto duplo sejam mais difíceis do que as tarefas de salto único, as tarefas de

salto triplo são tão difíceis quanto as de salto duplo. Ademais, as mudanças de perspectiva no eixo horizontal (x) são mais fáceis do que nos demais eixos (y e z).

Por fim, observam que atualmente os sistemas de ADD se concentram em mediar inimigos, a aparência de itens, a disponibilidade de recursos e outros detalhes de *design*. De toda forma, outra possível forma de mediação da dificuldade é a manipulação automática do *design* do nível por fatores incrementais.

3.1.2 Modelo estatístico

Koens apresentou um jogo de plataforma 2D com rolagem lateral, jogabilidade não-linear e caminhos bilaterais ao aplicar a Geração de Conteúdo Procedural (GCP) [8]. O método desenvolvido buscou encontrar um arranjo que possibilitava a realização de saltos e a satisfação de certas regras, sendo baseado em uma técnica de busca em profundidade. A dificuldade de cada nível gerado foi medida com base na presença de armadilhas que podem ocasionar a morte do jogador.

Foi construído um modelo estatístico para desafios individuais para medição de dificuldade, baseado em três parâmetros gerais do desafio:

1. Ciclo de tempo de uma armadilha;
2. Tempo requerido para o jogador mover do último ponto seguro para o primeiro com uma armadilha; e
3. Número de saltos necessários para o jogador atravessar o menor caminho do grafo.

O modelo, ilustrado na Figura 3.1, foi usado para encontrar o caminho que terá a maior probabilidade de sucesso ao atravessar o nível. Um experimento foi realizado com 53 voluntários para testar o modelo. Os resultados foram verificados investigando a relação entre a dificuldade computada e o número de mortes durante uma sessão de jogo.

A correspondência entre a quantidade de mortes e a dificuldade foi corroborada para participantes com experiência de jogo variada. Como sugestão para trabalhos futuros, indica gerar menos ruídos para os resultados com o desenvolvimento de níveis lineares em vez de uma multiplicidade de caminhos e de um modelo para computação da dificuldade do salto.

3.1.3 Medição da dificuldade de componentes individuais

A dificuldade em jogos de plataforma 2D gerados automaticamente foi o tema de pesquisa de Mourato e Santos em um estudo publicado em 2010 [1]. Um método de medição da



Figura 3.1: Exemplo de um grafo representante de um nível [8]. As arestas amarelas representam as rotas através do nível e as arestas brancas representam arestas arbitrárias.

dificuldade a partir da probabilidade de sucesso para completar um nível foi proposto para esse contexto.

O método utiliza a simplificação do que é categorizado como um obstáculo e não considera a presença de *checkpoints* ou vidas. Ademais, usaram o conceito de Compton et al [66] de níveis serem definidos como um ponto de início e um de fim, com alguns possuindo objetos que ferem o jogador. Em relação à experiência do jogador, considera que o usuário tem consciência do objetivo (chegar ao final da fase), que cada desafio é reconhecido e que tarefas triviais não são consideradas para a dificuldade do nível (como andar e subir escadas).

A medição da dificuldade de um nível do jogo analisa a probabilidade do jogador desistir, de repetir o desafio e de obter sucesso em um salto, sendo que a falha pode ocorrer ao levar dano e morrer ou ao ficar frustrado e desistir. Além disso, a medição da dificuldade de componentes individuais é feita para obter um modelo de probabilidade de sucesso do salto associada a cada componente, com base a fórmulas de física cinemática.

A combinação das probabilidades de sucesso de cada situação de salto resulta na estimativa de dificuldade do nível. A altura e a posição de uma plataforma, a distância entre plataformas e a posição dos inimigos podem ser parâmetros que determinam o sucesso do salto e facilitam ou dificultam o nível.

Um função de dificuldade indicando a probabilidade de sucesso $P(s)$ foi definida, sendo descrita pela Equação 3.1. A partir disso, jogos comerciais, como *Super Mario Bros.* e *Little Big Planet*, tiveram sua dificuldade mapeada e testada. Comparativamente, os resultados da dificuldade dos níveis foram mensurados e testados com participantes foi similar, observando que a dificuldade aumenta ao decorrer do jogo em ambos os jogos.

$$P(s) = P(j_1) \sum_{i=1} (P(j'_1) * P(r'))^i \quad (3.1)$$

$P(r)$ indica a probabilidade de renunciar após uma falha, sendo constante em um nível e relacionada à resiliência do jogador. $P(j_i)$ representa a probabilidade de alcançar sucesso em um salto ou em um desafio similar, com índice i .

Mourato mapeou margens de erro esperadas e a distribuição exponencial esperada para configurar a dificuldade em modelos de jogador. Para testar a teoria, os níveis foram segmentados manualmente, com base na interpretação visual das representações e foi determinada a vontade do jogador em continuar jogando pelo número máximo de tentativas para cada desafio.

Observa-se que não foi medida a probabilidade real de sucesso de um nível, mas estimada a dificuldade de distintos níveis, sendo possível compará-los. Além disso, esse estudo forneceu informações para a criação de mecanismos de GCP e para uma melhor compreensão sobre como determinar a dificuldade em um jogo de plataforma.

3.2 Inteligência Artificial

Além de vários outros campos nos quais a Inteligência Artificial (IA) pode ser aplicada, como saúde, finanças e manufatura, a indústria de jogos eletrônicos encontra aplicações complexas e extensas para esse conhecimento. Uma variedade de técnicas de IA são usadas em jogos eletrônicos para que o jogador possa ter uma experiência interessante e interativa [71]. Por exemplo, árvores de decisão, algoritmos de *pathfinding* e Rede Neural Artificial (RNA) podem ser usados para o comportamento inteligente de Personagem Não Jogável (NPC), a geração procedural de conteúdo e a caracterização do perfil do jogador.

Técnicas de Aprendizado de Máquina podem ser utilizadas para fornecer melhorias no comportamento dos agentes do jogo controlados pelo computador [77] e analisar o estado afetivo do jogador ou dados fisiológicos [9]. Facilita a geração e seleção automatizada de conteúdos e comportamentos, além de aprimorar agente de IA e criar novas possibilidades de engajamento em experiências dentro do jogo.

3.2.1 Aplicação de Inteligência Artificial em jogos

Árvores de decisão podem ser aplicadas para avaliação da motivação e do estado do jogador durante o jogo, permitindo a análise de informações comportamentais. Por outro lado, a análise dos dados de desempenho ou afetivos, a classificação da jogabilidade e posterior adaptação do fluxo do jogo podem ser feitas com um classificador *Naive Bayes*.

Blom et al. [9] comparou ambas as técnicas usando valores binários dos dados faciais dos jogadores em uma tarefa de classificação, usando *ZeroR* como método de linha de base.

Este estudo foi dividido em três partes metodológicas: a mensuração da dificuldade do jogo com base na análise da expressão facial, a personalização do jogo com base em heurísticas e em um modelo treinado. Observou-se que as expressões faciais, quando convertidas seus dados para o formato binário para análise, podem medir com precisão a dificuldade do jogo.

Um jogo de plataforma foi modificado para definir segmentos curtos a serem gerados, conforme mostrado na Figura 3.2. A dificuldade atual do jogo e uma estimativa do nível de desafio do jogador foram adicionados ao conjunto de recursos, juntamente com os dados afetivos. Esses dados ajudaram a discriminar entre atividade emocional espontânea e consistente.

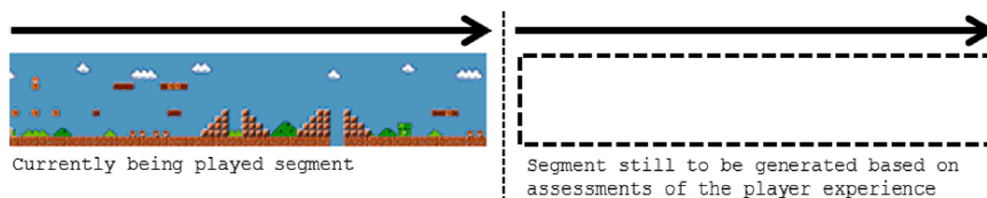


Figura 3.2: Configuração de jogo de plataforma para geração de conteúdo de jogo personalizado com base na classificação de expressões faciais [9].

Os métodos heurísticos forneceram uma base eficaz para atingir um estado afetivo apropriado com base nas emoções dos jogadores. Além disso, notou-se que um modelo de estado afetivo do jogador pode ser construído, mesmo com dados ruidosos causados pelo movimento da cabeça do jogador, conseguindo convergência mais rápida quando comparado ao método heurístico.

Nesse cenário, o *Naive Bayes* teve um desempenho inferior em comparação à classificação por meio da árvore de decisão, devido à suposição de que os recursos são independentes. As expressões faciais, mesmo que isoladas, são frequentemente correlacionadas e dependentes. Além disso, as mesmas técnicas podem ser utilizadas para criar um modelo probabilístico do estado afetivo dos jogadores, classificando suas emoções e prevendo a dificuldade percebida. A partir dessas informações, é possível adaptar o conteúdo gerado para o jogo de acordo apenas com os dados afetivos dos jogadores de forma discreta.

A lógica *fuzzy* é outra técnica que pode ser utilizada para a melhoria da jogabilidade, sendo utilizada por Araujo et al. [10] para investigar o uso do Ajuste Dinâmico de Dificuldade em robótica inteligente para o desenvolvimento de um sistema de treinamento de memória de trabalho. Embora a lógica seja aplicada em um robô, como visto na Figura 3.3, ela aplica as mesmas características dos jogos digitais, com o uso do robô motivado

pelo objetivo de manter a atenção dos jogadores enquanto trabalha com o sistema. Consideraram que uma interface amigável e interativa oferece uma experiência mais envolvente ao jogador e pode melhorar seu desempenho.

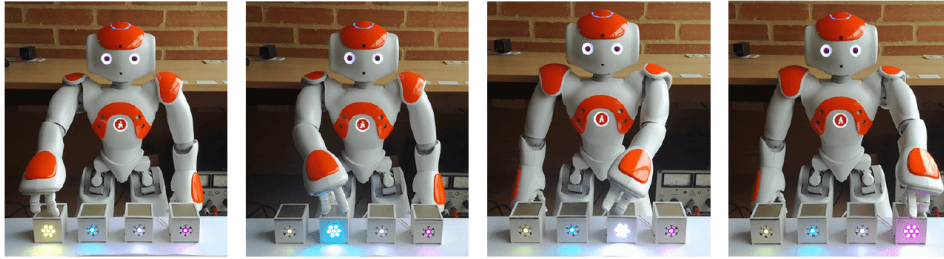


Figura 3.3: Configuração experimental para um jogo de treinamento de memória com um robô Nao movendo seus braços para gerar estímulos ao tocar em objetos inteligentes [10].

Neste contexto, uma tarefa *N-back* foi implementada com um sistema robótico. O jogador é apresentado a uma sequência de estímulos e deve se lembrar das sequências visuais, indicando quando o estímulo atual é o mesmo das etapas anteriores na sequência. Um agente de IA foi implementado para tomada de decisão e o ADD consistiu na geração de níveis customizados de acordo com um controlador *fuzzy*.

O sistema consiste em três componentes principais: usuário, robô e objetos inteligentes. A arquitetura BDI supervisionou o jogo *N-back* e o controle dos componentes do jogo, percebendo o ambiente por meio de sensores e tomando decisões de acordo com suas crenças mutantes do contexto. O sistema *fuzzy* foi integrado ao agente BDI inteligente como parte do processo para gerar uma ação final e equilibrar os níveis do jogo.

3.2.2 Adaptatividade com Aprendizado de Máquina

A adaptação dos conteúdos do jogo, a avaliação do comportamento dos jogadores e a previsão dos caminhos podem ser computados com Busca em Árvore Monte Carlo (MCTS). Por exemplo, com um conjunto de níveis ou segmentos existentes, o sistema pode aprender padrões estatísticos e decidir como gerar um nível. Essa decisão pode garantir a criação de níveis jogáveis, de acordo com o conteúdo gerado anteriormente e com os dados de desempenho do jogador.

Uma série de estudos usa um agente de IA ajustado dinamicamente com base no MCTS. Esse uso é especialmente proeminente no contexto de jogos de luta, como *FightingICE*, uma plataforma de jogo de luta 2D em tempo real onde 2 jogadores se enfrentam em uma arena usando uma combinação de ações (mover, pular, agachar, bloquear, socar, chutar e ataques especiais) para diminuir a saúde do oponente. Joga-se em rodadas

cronometradas e termina quando o tempo é atingido ou a saúde de um dos personagens chega a zero.

Ishihara et al. [72] visa jogadores iniciantes e intermediários, para manter o agente de IA com um comportamento confiável e para entreter e melhorar a habilidade dos jogadores. Nesse sentido, visa que a IA tenha ações coerentes, de forma que não prejudique a motivação dos jogadores, como evitar sofrer danos de forma intencional e sem resistência. O ADD proposto para o agente de IA poderia ajustar sua força ao oponente humano e restringir ações não naturais, especialmente para jogadores intermediários.

O agente de IA conduz a ação com base na situação atual do jogo enquanto visa o ajuste para lutar uniformemente com uma pequena diferença na proporção de vitórias. Por exemplo, executa ações fortes ao perder e ações fracas ao vencer. A lógica básica faz com que a repetição de ataques sem golpe e o agente se afaste do jogador mesmo quando eles já estão distantes.

Comportamentos agressivos visando derrotar um oponente, como acertar os ataques de forma adequada, são a definição de comportamento verossímil para o propósito do jogo de luta [72]. Por outro lado, o que é considerado contrário a esse propósito, como ataques no-hit, define comportamentos não naturais. Em suma, as ações de IA selecionadas consideram o ajuste da dificuldade em relação à habilidade do oponente e as formas de derrotá-lo.

Considerando a imersão do jogador, Demediuk et al. propôs novos agentes de IA ajustados dinamicamente [78]. Os agentes propostos fornecem um nível de dificuldade adaptado de acordo com a habilidade dos jogadores, alterando a política de seleção de ações e a avaliação do MCTS em tempo real. Testar agentes de IA uns contra os outros e contra jogadores humanos mostrou que esses novos agentes de IA são capazes de ajustar o nível de dificuldade em tempo real quando objetivam uma diferença zero de saúde como resultado do jogo.

Em outro estudo, Demediuk et al. [79] notou que medir a Média do Valor Heurístico dos jogadores pode levar a obter a mesma classificação dos jogadores de sistemas de última geração sem competição direta. Este método torna o ADD adequado para contextos educacionais e de treinamento.

Além disso, o método proposto de agentes de IA ajustados dinamicamente com MCTS [72] foi aplicado para uma versão de corpo inteiro do jogo *FightingICE* [80], na qual o jogador utiliza movimentos corporais em vez de um controle para jogar. A IA dos jogos de movimento encorajou exercícios bem equilibrados ao prever a probabilidade das ações de contra-ataque dos jogadores de acordo com as ações feitas pela IA. Este tipo de mecanismo pode aumentar o equilíbrio, o prazer e a gratificação quando comparado a um IA de MCTS típico.

Por sua vez, redes neurais podem ser utilizadas para melhorar as propostas de adaptabilidade em jogos. Nesse sentido, Fujita [81] aplicou uma Rede Neural Artificial profunda e um MCTS para estimar o valor do estado de jogo a partir do estado de tabuleiro, testado nos jogos Othello e Connect4. Com isso, adaptou a dificuldade da IA oponente, balanceando de acordo com a habilidade do jogador percebida pelo estado do jogo. Indicam que o método pode ser aplicado em qualquer jogo que possa ter seu valor estimado a partir de seu estado.

Or et al [82] propuseram um método de ADD que otimizava a experiência do usuário ao considerar outros jogadores e regras do jogo. Baseou-se em uma Rede Neural Artificial profunda que diminuía a perda da função de experiência do usuário de acordo com dados do usuário e de outros similares.

Experimentos online foram feitos com mais de 200 participantes em um jogo de quebra-cabeça, superando as heurísticas manuais feitas por designers do jogo. Possui um contraponto de não incorporar dados em tempo real, podendo não corresponder à dificuldade ao jogador durante o período inicial de definição da dificuldade para o usuário. Ainda assim, o sistema é adaptável para outros jogos, precisando apenas de um método de definição de dificuldade para os dados do jogo e uma definição de similaridade entre jogadores.

O estudo de Yannakakis et al [52] ajustou parâmetros de um jogo com interação física da *Playware* chamado *Bug-Smasher*. As adaptações examinam o nível de satisfação reportado ao construir uma função de mapeamento do jogo e do usuário por meio de redes neurais. As variáveis utilizadas consideram o tempo de resposta, a pressão do pé, o número de interações e a curiosidade e diversidade do inimigo.

Para o jogo *Bug-Smasher*, obteve-se uma acurácia de 77.77% ao se basear em modelos e estudos prévios sobre a preferência de entretenimento de crianças, o público-alvo da pesquisa. Testes foram realizados com a versão original e a adaptada, questionando qual era a mais divertida. Houve um sucesso na adaptação do modelo devido ao aumento do valor de entretenimento em 75,67% para o jogo adaptado, indicado pelo número de interações, pelo tempo de resposta e por uma métrica de curiosidade dos participantes.

3.3 ADD a partir do desempenho do jogador

A área do ADD com maior proliferação se refere ao ajuste por meio do desempenho, estando vinculado aos componentes que são adaptados e quais os parâmetros para determinar o desempenho do jogador. De toda forma, são inúmeros os métodos dentro de cada gênero analisado, dependendo dos objetos examinados e dos objetivos de cada estudo.

3.3.1 Estado da Arte

Uma revisão das abordagens atuais do ADD [83] foi publicada em 2018, apresentando uma análise sobre a área e suas direções em diversos tipos de jogos na última década, e destacando alguns dos tipos mais representativos para cada aplicação. Constatou-se que, após 2009, tem sido realizadas muitas pesquisas relacionadas a métodos para desenvolver ou melhorar o ADD, sendo que a quantidade de pesquisas cresceu na década anterior ao estudo e que houve quase três vezes mais trabalhos de pesquisa em 2012 e 2017 (anos de pico no estudo) ao comparar a 2009.

Atribuem um valor para qualquer estado de jogo que seja indicativo do nível de dificuldade sentido pelo jogador a qualquer momento. Com isso, existem vários caminhos para classificar as abordagens, sendo descritas no estudo as seguintes:

1. Métodos probabilísticos.
2. *Perceptrons* Simples e Multicamadas, contendo estudos sobre jogos de plataforma, mais especificamente associando Geração de Conteúdo Procedural (GCP), Ajuste Dinâmico de Dificuldade (ADD) e Rede Neural Artificial (RNA) (mais detalhes na Subseção 3.5.1);
3. Códigos dinâmicos;
4. Sistema de Hamlet;
5. Aprendizagem por Reforço;
6. Confiança Superior Limitada por Árvores e Redes Neurais Artificiais;
7. Sistema Auto-organizado e Redes Neurais Artificiais.

O artigo analisa vários métodos para ADD que são propostos na literatura. O aspecto comum em todos os métodos é um requisito para medir, de modo implícito ou explícito, o nível de dificuldade enfrentado pelo jogador em qualquer instante, estimado por funções heurísticas. Considerando a literatura analisada, observam a necessidade de pesquisas futuras em:

- Novas técnicas baseadas em busca para identificar níveis ótimos;
- Modelos de jogadores, indo além dos tipos de agente;
- Aumentar a complexidade pela adição de mais variáveis que considerem outros aspectos do jogo;
- Investigar a possibilidade de cobrir características como estilo de jogo;

- O conceito de mapear o jogador humano e, de acordo com isso, desenvolver um modelo de jogador;
- Um modelo de jogador que inclua mais aspectos comportamentais;
- Integração de técnicas de modelagens de jogador e abordagens atuais do ADD.

Apontam que as técnicas de ADD foram corroboradas na literatura como sendo ferramentas úteis para incorporação em sistemas complexos e dinâmicos. Ademais, indicam que existem inúmeras aplicações para estudos de ADD em vários domínios, incluindo generalizações e extensões.

3.3.2 Modelagem do jogador

Araújo realizou um estudo sobre modelagem do jogador [20]. A partir de um FPS 2D, analisa a influência da adaptabilidade na experiência de jogo sob a ótica da teoria de fluxo [51] e avalia o modelo proposto pelo método CEGE [84]. Nesse cenário, as variáveis do inimigo (velocidade, tempo de tiro e alcance) são modificadas conforme a dificuldade, havendo um número finito de vidas para o jogador e dificuldades pré-definidas.

O trabalho constatou que mudanças mais fáceis de perceber impactam negativamente a experiência do jogador. Nesse sentido, analisaram jogos já estabelecidos no mercado, como *Mario Kart* e *Oblivion*, e testados no meio acadêmico, como *Infinite Adaptive Mario*. Observou-se que um jogador com desempenho ruim pode se sentir desmotivado quando as mudanças no jogo são evidentes ou quando compreende como o sistema de ajuste funciona. Além disso, torna-se possível a quebra de imersão e o abuso do sistema pelo jogador.

As versões com e sem adaptatividade foram testadas com questionários prévios e ao final do jogo, procurando avaliar a percepção da adaptatividade e demais dados qualitativos. A detecção de diferença real na dificuldade foi baixa (8%) e jogadores dedicados tiveram uma menor frustração com a versão adaptada, porém o contrário ocorreu com jogadores casuais, justificado pela escolha do gênero *shoot'em up*.

As contribuições do trabalho foram: a implementação do modelo de Charles et al [85], uma versão adaptativa eficiente do jogo, definições de conceitos (como de jogos parcialmente adaptativos) e a comparação do resultado entre jogadores casuais e dedicados. Sugeriram o refinamento do *gameplay*, a maior familiarização antes de realizar os testes, uma maior discretização dos níveis de dificuldade e testes com outros gêneros e técnicas.

3.3.3 Distribuição de recursos e desafios

Hintze et al [23] testaram uma nova abordagem para ADD em um jogo 2D no qual o jogador deve fugir de um agente predador por um tempo específico, com o suporte de agentes colaboradores. Uma Inteligência Artificial foi evoluída ortogonalmente tanto para os colaboradores quanto para os oponentes, tendo incentivos ortogonais e possibilitando um ajuste mais granular.

A adaptação foi realizada por meio da distribuição de oponentes e recursos, modificando as técnicas dos oponentes. Enquanto isso, a interação humana pode ser observada pelo desempenho e comportamento dos agentes colaboradores. A evolução ortogonalizada é uma área promissora, conforme suaviza o ajuste da dificuldade e permite novas experiências no jogo.

Christyowidiasmoro et al [86] trabalharam com o jogo de estratégia *Tower Defense*, com um lado buscando destruir a torre do outro. Os autores propuseram uma fórmula para descobrir a dificuldade do jogo e se o balanceamento está adequado.

A dificuldade é baseada nos recursos recebidos (quanto maior, mais recursos os oponentes recebem para construir suas unidades por tempo). Para isso, três dificuldades pré-determinadas foram implementadas na Inteligência Artificial do oponente e são especificadas pelos pontos de vida totais de todas as unidades, o ataque total das unidades e quantas unidades e recursos o jogador possui.

Foram realizados testes nos quais ambos os lados eram controlados pela IA, combinando as permutações entre as três dificuldades. Constataram a validade da fórmula proposta para medir a dificuldade e mostrar o equilíbrio entre os níveis de dificuldade, viabilizando a aplicação em outros gêneros. Todavia, a proposta se restringe a três níveis de dificuldade definidos e a jogos que a distribuição de recursos é parte essencial da jogabilidade.

3.3.4 Compatibilidade do desafio

Hunicke [24] analisou a importância do jogador não perceber o Ajuste Dinâmico de Dificuldade em uma versão modificada do jogo FPS *Half Life*. Para manter o jogador em estado de fluxo, analisa a mudança em conteúdos básicos do design, como a demanda e o fornecimento de itens, e averigua como isso afeta a percepção da dificuldade e a experiência do jogador. Utilizaram uma mecânica baseada em diagramas de estado de um jogo do gênero analisado para balancear, mantendo o progresso do jogador e os objetivos fundamentais do *design* do jogo.

20 participantes jogaram uma sessão de 15 minutos e responderam um questionário acerca da percepção de mudanças e da dificuldade. Demonstrou-se que jogadores expe-

rientes tinham um maior aproveitamento e havia uma queda na taxa de morte com o ajuste. Ainda assim, o artigo indica que não foi conclusivo para informar se a dificuldade ajustada traz uma melhoria para a experiência do jogador.

O estudo de Denisova et al para jogo do gênero FPS [22], assim como o anterior, procura executar o ADD sem que o jogador tenha conhecimento. Por outro lado, utilizaram a manipulação do tempo como método de adaptação. Mais tempo para terminar o desafio para jogadores com menor desempenho e vice-versa, em vez de adaptar personagens não-jogáveis ou componentes interativos.

Os autores almejavam comprovar a tese de que balancear o desafio do jogo conforme a habilidade do jogador levaria a uma melhora na experiência do jogador e promoveria imersão para todos. Com isso, testaram o jogo com um grupo de participantes com e sem a adaptação. Por meio da pontuação obtida, da resposta de questionários e da utilização de métricas de análise de variância, mostraram que os jogadores consideraram o jogo mais imersivo quando o tempo era manipulado.

3.3.5 Perfil do jogador

O balanceamento dinâmico da dificuldade em um jogo de quebra-cabeça foi o tema de estudo de Hawkins et al [11]. Buscou-se atender diferentes níveis de habilidade e manter a satisfação de acordo com as características do perfil do jogador, dependendo do gosto por desafios e riscos.

Uma técnica de modelagem conhecida como filtragem de partículas foi utilizada, sendo aplicada para modelar vários níveis de habilidade e considerar o perfil de risco do jogador. Assim, procuraram definir o perfil do risco entre os que se arriscam (tomando decisões rápidas com menor chance de sucesso) e os que são cautelosos (esperando ter certeza em detrimento do tempo gasto). A Figura 3.4 ilustra os parâmetros utilizados para ajustar os graus de similaridade entre o que é ou não alvo, movendo diferentes perfis para performance alvo.

Nenhuma dessas propriedades psicológicas subjacentes sobre o perfil de risco pode ser inferida a partir da observação crua da precisão e do tempo de resposta. Em vez de desenvolver métodos que combinassem as duas medidas em algum composto, importaram uma teoria de decisão da psicologia cognitiva que apoia a estimativa rápida e eficiente das saídas esperadas.

A técnica foi demonstrada em um desafio no qual os jogadores são obrigados a tomar uma decisão entre um número de alternativas possíveis, sendo que apenas uma está correta. Ao coletar dados empíricos para o tempo de resposta e a precisão do jogador, desenvolveram o supracitado modelo de filtro de partículas. Nesse caso, os filtros se re-

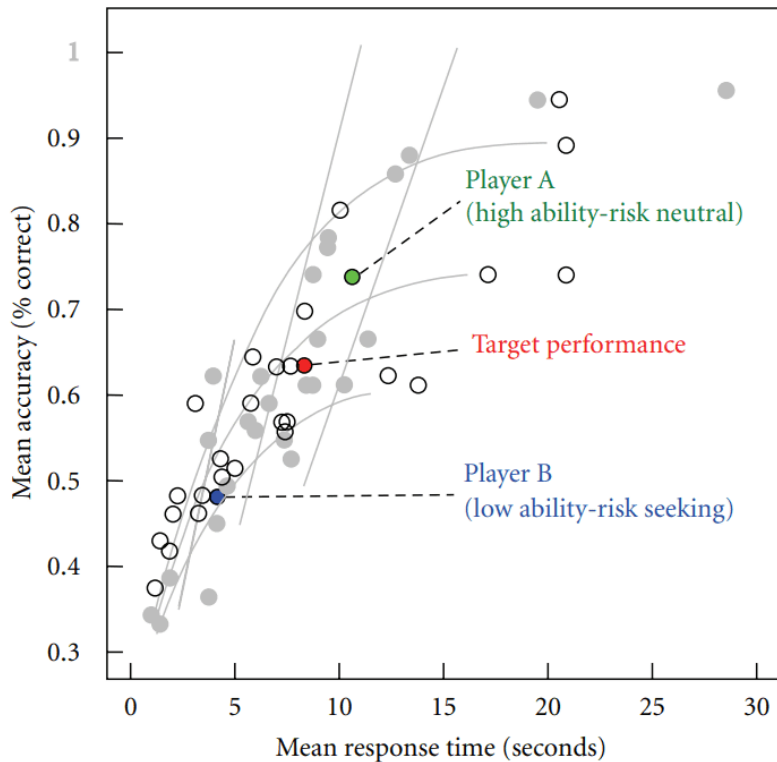


Figura 3.4: Habilidade e perfil de risco do jogador medidos dinamicamente para identificar tamanho do filtro de partícula e limiar de decisão [11].

lacionam ao desafio ocasionado por uma decisão, desenvolvendo modelos de jogadores dinâmicos.

O estudo de Nagle et al [87] explorou a relação entre a personalidade e mecânicas de jogo. Investigou a relação entre perfis de jogador e quatro diferentes modelos de ADD em um jogo FPS. Assim, um modelo de regressão linear foi aplicado para relacionar os perfis do modelo Big Five [58] com o prazer e a duração da sessão de jogo, predizendo a adaptação da dificuldade para maximizar esses dois fatores. O preditor foi testado comparando com modelos de ADD e tiveram maior acurácia para identificar grupos. Ressaltaram que a importância da adaptação da dificuldade e que a personalidade do jogador pode estar vinculada a outras mecânicas de jogo importantes, propondo um *framework* de customização de jogo baseado no perfil do jogador como trabalho futuro.

3.4 ADD afetivo

Ainda que o ADD afetivo tenha uma menor proliferação comparado ao ADD por desempenho, a área continua crescendo e demonstrando resultados que corroboram a utilização de dados afetivos para modificar a experiência do jogo.

3.4.1 Estado da arte

Uma revisão da literatura da área de ajustes afetivos foi elaborada por Bontchev e publicada em 2016 [5]. Analisa-se modelos para apresentação de emoções, técnicas de mensuração de sinais comportamentais, reconhecimento de emoções e mecanismos de adaptação aplicados em videogames com resposta afetiva. Ao final, foram selecionados 14 estudos, a partir de uma base inicial de mais de 300. Como critério, foi estabelecido que os estudos deveriam ter sido desenvolvidos na década antecedente à revisão, refletirem métodos adaptativos e afetivos, possuírem resultados experimentais e validação, além de serem escritos em inglês.

Somente dois jogos foram desenvolvidos com finalidade aplicada, sendo um com foco cognição de equipe, comunicação e coordenação, e outro para estudos clínicos de nível de concentração. O gênero dos jogos é diverso, sendo variações de jogos *arcade* antigos (como *Pong* e *Pac-Man*) ou modificações de jogos populares de FPS, plataforma ou corrida. Todos foram desenvolvidos para computador, em três casos com controle externo, e somente dois são para múltiplos jogadores.

As adaptações variam fatores específicos do jogo, como mudanças áudio-visuais e propriedades de oponentes. Os mecanismos de adaptação incluem principalmente *feedback* afetivo positivo, aumentando o desempenho, a diversão e a aleatoriedade do jogo, ou *feedback* negativo, criando estabilidade comportamental. Alguns casos usam algoritmos adaptativos mais complexos para manter o jogador no estado de fluxo, mudando de *biofeedback* positivo para negativo e vice-versa.

As adaptações afetivas são realizadas principalmente usando sinais de Sistema Nervoso Autônomo (SNA) obtidos por Eletrocardiografia (ECG), Fotopletismografia (PPG), Resposta Galvânica da Pele (GSR), Eletromiografia (EMG) ou medindo os ritmos de respiração, temperatura ou pressão do teclado, com apenas dois casos utilizando o Eletroencefalograma (EEG). As medidas do SNA são mais usadas em jogos afetivos do que os sinais do Sistema Nervoso Central (SNC) por representarem as dimensões emocionais em vez de emoções discretas. No caso, os sinais do SNC são usados para extrair um número relativamente pequeno de características (2 ou 3) em comparação aos do SNA (1 a 17).

A janela de tempo para os sinais analisados difere de 0,025s a 180s e depende do sinal medido, do recurso extraído e do propósito de adaptação. A adoção de uma janela de tempo muito curta aumenta os riscos de influência de flutuações repentinas no valor do sinal medido. Enquanto isso, a escolha de uma janela de tempo longa reduz a sensibilidade do laço adaptativo.

Em onze dos estudos, os sinais brutos são filtrados para remover o ruído e para separar bandas de alta e baixa frequência. As características do sinal extraído são usadas para a inferência de emoções (como estresse, frustração, ansiedade, tédio, diversão e excitação)

ou para mapeamento direto das características adaptadas no jogo, o que é realizado por metade dos estudos.

Os experimentos variam de 5 a 75 sujeitos participantes, com uma maior participação masculina, idades que variam entre 8 e 10 anos (valores mínimos) e 18 e 54 anos (valores máximos), e sessões que duram entre 1,5 e 10 minutos. Por sua vez, os resultados obtidos pelos questionários demonstram que jogos adaptativos criam mais diversão nos jogadores do que os de dificuldade estática e trazem um melhor desempenho da maioria dos participantes. Além disso, a jogabilidade com o uso da afetividade se torna mais desafiadora e satisfatória do que em um ADD baseado somente no desempenho.

3.4.2 Análise em Tempo Real

Liu et al [25] propuseram reconhecer os estados afetivos do jogador e investigar o uso desses para ajustar a dificuldade em tempo real a partir do jogo *Pong* e de um jogo de anagramas. Primeiramente, os sensores fisiológicos capturaram dados (ECG, EDA, BVP, EMG e ICG) durante a sessão do jogo, estimando a ansiedade do jogador. Questionários foram aplicados em intervalos de tempo procurando reconhecer o estado afetivo de ansiedade dos jogadores.

Após isso, utilizaram versões adaptadas do jogo *Pong* para verificar o impacto da sensibilidade da afetividade na experiência do jogo. As versões constituíram no uso de ADD baseado no desempenho e no nível de ansiedade do jogador, comparando-as por meio de dados do jogo e de questionários prévios e posteriores às sessões de teste.

Os modelos afetivos tiveram alto índice de acurácia, com a performance melhorada, a atividade percebida como mais desafiadora e satisfatória, e o nível de ansiedade reduzido na maioria dos voluntários. Assim, fortaleceu-se o argumento do que a experiência de um jogo pode ser melhorada com o ajuste de variáveis afetivas em tempo real.

Imre [27] concentrou-se na área de ADD afetivo por meio da resposta de sensores, objetivando alcançar um estado de fluxo. Procurou demonstrar que a análise de condução por meio da pele pode ser usada para ajustar o jogo em níveis pré-determinados de dificuldade.

Executou-se uma ampla avaliação dos métodos de Ajuste Dinâmico de Dificuldade, da modulação de dificuldade e da confiabilidade da Atividade Eletrodérmica para medição de estresse e da carga cognitiva. Com isso, desenvolveu e testou o jogo FPS 2D *Electroderma* e corroborou que a condutância da pele pode ser usada como variável da modulação de dificuldade em um ADD.

Constatou que o ADD pela condução de energia pela pele pode funcionar melhor ao associar com o batimento cardíaco. Como desafios futuros, aponta que o uso de variáveis não controladas do mundo exterior podem influenciar o jogador, também sugere a análise

da usabilidade dos sensores e indica a necessidade da padronização de softwares relativos à afetividade.

Um jogo de terror foi criado por Moschovitis et al [88] para aumentar a dificuldade quando o jogador está estressado e diminuir quando está relaxado a partir de dados biométricos. Foram alteradas velocidade, aceleração e tamanho da área para identificação por inimigos. Analisaram o impacto dessa abordagem nos desafios cognitivo, emocional, de performance e de escolha de decisões, comparando com o jogo sem ADD.

Resultados mostraram que os jogadores se sentiram mais motivado ao adaptar o jogo pela frequência cardíaca. Todavia, dentre os quatro tipos de desafios, apenas a escolha de decisões foi afetada pelo uso do ADD.

3.4.3 Afetividade associada ao desempenho do jogador

Considerando que a maioria dos métodos de Ajuste Dinâmico de Dificuldade estudados referem-se ao ajuste por meio do desempenho, Aguiar e Fernandes [43] desenvolveram um mecanismo de ADD híbrido. Os autores investigaram se esse mecanismo faz o jogador ter uma melhor percepção do estado de fluxo, comparando com modelos que utilizam apenas uma das variáveis.

O jogo FPS 2D *Asteroids: in the 2nd and 1/2th Dimension* foi adaptado para utilizar um sensor da Atividade Eletrodérmica e coletar dados fisiológicos do jogador. A dificuldade é ajustada por intermédio da margem de velocidade e a densidade dos asteroides, variando entre três conjuntos de valores pré-determinados.

Em uma primeira bateria com 14 voluntários, foram testadas a dificuldade e a imersão do jogo por meio de um questionário respondido ao final da sessão. Com esses dados, associados aos captados pelo jogo e pelos sensores, foram definidas as margens para alterar a dificuldade, modificando-a conforme a taxa de morte do jogador e seu estado afetivo.

Em uma segunda bateria, os modelos de ADD (baseado em desempenho, em dados fisiológicos e híbrido) foram testados com 16 participantes. Ao final da sessão, um questionário analisou a compreensão do objetivo, o desafio percebido, a imersão e os dados sociodemográficos. A média das respostas dos participantes que jogaram o modelo híbrido foi maior do que a relacionada aos outros dois modelos baseados em apenas uma variável, sugerindo uma averiguação mais ampla e com outros sensores.

Natal [89] investigou os componentes que podem melhorar a chance do jogador ser levado ao estado de fluxo com o modelo híbrido, utilizando como base o estudo de Aguiar e Fernandes. Testes foram conduzidos almejando determinar as variáveis relevantes para implementar o ADD, sendo essas a velocidade e a densidade. Em seguida, buscou confirmar se o jogador entrou em estado de fluxo com e sem o uso dos dados afetivos no ADD a partir de questionários.

Observou-se que, atuando de forma singular, a velocidade dos asteroides possui maior relevância para a percepção da dificuldade e que, ao atuarem em conjunto, possuem uma influência maior em comparação a quando não estão combinadas. Para trabalhos futuros, sugere o teste com novas ferramentas e variáveis.

3.5 ADD em jogos de plataforma

Estudos analisaram a aplicação do ADD em jogos de plataforma, existindo diversas abordagens para adequar o desafio à habilidade do jogador. Entretanto, cada método possui suas especificidades e acrescentam novas considerações para a área.

3.5.1 Geração procedural de níveis

O ADD está presente em jogos de plataforma especialmente quando associados com a Geração de Conteúdo Procedural (GCP). Desse modo, os níveis são adequados ao jogador enquanto gerados por algoritmos em vez de adaptar níveis criados previamente.

Modelos computacionais da experiência do jogador

Segmentos de nível são gerados a partir de modelos computacionais da experiência de um jogador derivados da interação no jogo no estudo de Pedersen et al [90]. Para isso, utilizaram dados coletados e aprendizado de máquina para criar uma curva de dificuldade crescente que é adaptada conforme o jogador morre em uma versão modificada do jogo *Infinite Mario*.

Analisou-se a relação entre os parâmetros de design de níveis de jogos de plataforma (tamanho, posicionamento e diversidade espacial de buracos e a presença de mudanças de direção), a experiência do jogador e as características individuais do jogo. Além disso, a diversão, o desafio e a frustração foram tidos como os constituintes da experiência de um jogador.

A partir da análise de dados dos testes, os autores tentaram elaborar uma aproximação de função baseada em jogabilidade e características controláveis para registrar as escolhas emocionais utilizando a aprendizagem de preferência neuro-evolutiva [91]. Todavia, os dados representando a função estavam com ruído, pois as escolhas dos jogadores eram subjetivas e o modo de jogar variava.

Devido a problemática apresentada e por ser uma função não-linear, uma Rede Neural Artificial (RNA) foi selecionada para a aproximação no mapeamento entre dados e emoções relatadas. Após uma validação por testes online, um algoritmo genético foi implantado, utilizando uma função que media a variação entre as preferências emocio-

nais registradas em questionários e o resultado do modelo correspondente. Os resultados mostraram que houve alta precisão de predição de diversão (69,18%), desafio (77,77%) e frustração (88,66%).

O estudo teve continuidade em [92], onde apresentaram os resultados e o desempenho da construção dos perceptrons multicamadas (MLP) e discutiram as maneiras pelas quais os modelos induzidos poderiam gerar o conteúdo do jogo automaticamente.

Shaker et al [93] desenvolveu um modelo a partir dos trabalhos de Pedersen et al [90][92] para geração automática de níveis personalizados em uma versão modificada do jogo *Infinite Mario*. Para gerar conteúdo personalizado de acordo com a experiência do jogador em tempo real, foi considerado necessário prever emoções a partir de recursos controláveis. Assim, foi utilizada uma lógica adaptada de perceptrons de camada única para avaliar aproximadamente o nível afetivo dos jogadores.

Os autores analisaram a adaptação dinâmica às mudanças no modo de jogar e a capacidade do modelo de generalizar jogadores de vários tipos. Mobilizaram dois agentes de Inteligência Artificial (IA) para jogar em turnos e monitorar o crescimento do valor de diversão em um nível gerado aleatoriamente. Realizaram o mesmo teste com quatro humanos e compararam os resultados de diversão e performance com os modelos propostos. Em vista dos resultados terem sido similares, mostraram a capacidade do modelo de se ajustar ao jogador.

Padrões reais de jogo

Alegando que a maioria dos métodos de ADD é baseada em intuições dos designers em vez de padrões reais, Jennings-Teats et al [94][95] desenvolveram um jogo de plataforma 2D denominado *Polymorph*. O jogo utiliza métodos de aprendizado de máquina e geração de nível para analisar a habilidade do jogador e o nível de dificuldade, criando de forma dinâmica níveis com desafios continuamente desejados.

O estudo valida se o jogo consegue alcançar um ADD otimizado e se o jogador está com dificuldade antes de morrer. A dificuldade do segmento que será gerado é definida pelo análise do tempo parado ou andando para trás, do tempo para completar o nível e do número de mortes.

Perceptrons multicamadas acessados a partir de traços de jogo foram utilizados, os quais são reunidos usando uma ferramenta online que atribui aos usuários vários componentes e os classifica em um nível de dificuldade. Após isso, os modelos de dificuldade são usados para escolher o segmento de nível adequado para o desempenho atual do jogador.

Para validação, foram executados testes online, analisando a correlação de um conjunto de dois componentes do jogo, como salto e buraco ou buraco e buraco, com a dificuldade percebida pelo jogador.

3.5.2 Dificuldade associada à geração dinâmica adaptativa

Após os estudos realizados em [1] e [96], sendo descrito na Seção 3.1.3, Mourato dissertou sobre a melhoria dos processos de Geração de Conteúdo Procedural em jogos de plataforma [12]. Conforme as técnicas da área evoluíram, algoritmos mais ambiciosos e complexos surgiram, indo além da pesquisa por geometrias válidas para servir de base a um nível. Assim, abrangeu-se características de *design* direcionadas ao usuário e detalhes criativos normalmente relacionados à criação humana.

No campo de GCP, iniciou-se a investigação de novas técnicas que aplicassem padrões de *game design* e implementou-se um algoritmo para geração de estruturas globais e outro para adaptação do conteúdo. Com isso, foi elaborada a transformação de caminhos simples em desafios com estruturas complexas, alterando componentes em locais estratégicos. Ou seja, o método executava a adição ou retirada de oponentes, armadilhas, poderes e/ou vida; a alteração no tamanho de buracos; a criação de desvios pela necessidade de algum item ou ação (como pisar em uma plataforma) para prosseguir; e a definição de caminhos cooperativos para dois jogadores. Esse padrão está ilustrado na Figura 3.5.

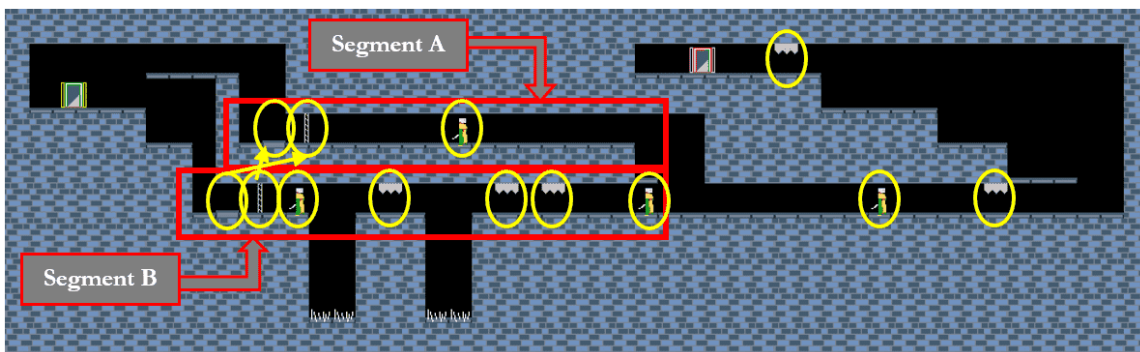


Figura 3.5: Exemplo de um nível com dois segmentos independentes, ajustado individualmente para dois jogadores de diferentes habilidades [12].

A adaptação do conteúdo analisa padrões de sucesso e falha, sendo feitas com base no tamanho do percurso, na dificuldade desejada e na estimativa do estado do jogador em cada vértice do grafo. Esse tipo de abordagem evita níveis lineares e potencializa a adição de padrões com conteúdo mais dinâmico. Além disso, foi validado em testes com 40 voluntários ao verificar estatisticamente as dificuldades mensuradas e as obtidas em sessões do jogo.

Em complemento, foi desenvolvido um *framework* para representação do conteúdo de níveis de jogos de plataforma, visando a integração de diversas técnicas. A abordagem é genérica para não depender do jogo considerado e segue uma arquitetura modular para a geração. Além disso, elaborou-se um editor de níveis padrão e outro semi-automático, a partir de missões e objetivos que geram uma estrutura física correspondente.

Conforme apresentado na Figura 3.6, um protótipo que integrasse as distintas contribuições foi implementado e testado, reforçando a validade da arquitetura e do *framework* propostos e colhendo dados para corroborar a validação das métricas relativas à percepção da dificuldade. Os testes foram realizados com 93 jogadores, em 52 níveis autorais e 26 gerados, totalizando 1587 tentativas.



Figura 3.6: Captura de tela do jogo protótipo usado em [12].

Em cada teste o usuário recebia um nível aleatório, o avaliava em uma escala e especulava se foi gerado por um humano ou computador. Os participantes tiveram dificuldade em diferenciar os níveis criados por um designer e os gerados automaticamente, possuindo avaliações similares. Em relação a estimativa da dificuldade feita pela probabilidade de sucesso, comparou-se as taxas efetivas de sucesso e falha dos jogadores participantes e os valores estimados para cada componente considerando diferentes características espaço-temporais, obtendo uma taxa de erro em torno de 10%.

3.5.3 Abordagens para personalização de níveis

Diversas versões aprimoradas do jogo *Infinite Mario* foram desenvolvidas utilizando dados de desempenho do jogador para gerar adaptações na geração dos níveis. Uma abordagem para personalizar o espaço dos níveis de um jogo de plataforma foi proposta por Bakkes et al [97], balanceando para diferentes tipos de usuário. Considera-se nesse estudo dois desafios de *design*: o risco de abandono e o *feedback* implícito do usuário.

Por sua vez, outros artigos concentraram-se na estruturação de primitiva de construção [98] ou em manter o usuário no estado de fluxo [99]. No último caso, estudaram quais parâmetros devem ser levados em consideração para determinar a dificuldade. Utilizaram limiares não-fixos e automatizados para cada jogador, alterando coletáveis ao verificar a variação da dificuldade em ciclos temporais definidos.

Seguindo o padrão do jogo *Mario*, Wheat [100] desenvolveu um jogo de plataforma simples que utiliza de computação evolucionária para geração procedural e adaptação dinâmica. Por intermédio de um algoritmo genético, analisou-se uma lista de regras para identificar a habilidade de agentes, verificando o enquadramento em alguma dessas a cada ciclo temporal. As regras foram feitas para priorizar chegar ao fim do nível, evitar obstáculos e coletar o máximo de pontos.

Esse sistema foi testado por 17 participantes, os quais jogaram um conjunto de níveis gerados e responderam questionários sobre a diversão e a dificuldade experienciadas. Observou-se que uma maior aptidão é alcançada após numerosas gerações do algoritmo genético. Entretanto, foi impraticável que os participantes jogassem todos os níveis de solução no ciclo evolutivo.

3.5.4 Dados fisiológicos na adaptação de jogos de plataforma

Vollmers [28] conduziu um estudo objetivando projetar e implementar um produto que eliminasse fontes de frustração em um jogo e elevasse a experiência geral. Para descobrir como desenvolver um sistema de ADD para jogos de plataforma que use dados fisiológicos como entrada, analisou o que torna um jogo difícil e desafiador e como esses parâmetros podem ser ajustados.

Almejou conferir se o ADD com dados fisiológicos consegue manter o nível de frustração do jogador em uma faixa considerada adequada. No trabalho foi utilizado o *Empatica Wrist Band* para captar os níveis de frustração por meio da Resposta Galvânica da Pele. Todavia, a dificuldade era ajustada de forma manual por um pesquisador que monitorava os dados do jogador. Ou seja, adaptava a vida do inimigo quando o jogador ficava frustrado, com o pesquisador diminuindo manualmente a vida dos inimigos.

Foram realizados testes para verificar a usabilidade do aparelho e o ajuste dinâmico do jogo, até que o usuário completasse o nível ou que alcançasse um tempo limite. Mesmo que o resultado tenha sido efetivo para o ajuste e a experiência dos usuários, foi realizado com um contingente pequeno de participantes e o resultado estava enviesado pela aplicação do ADD ser executada manualmente em vez de um algoritmo.

Capítulo 4

Contextualização

Este capítulo apresenta a contextualização do problema, as ferramentas e os equipamentos utilizados, o jogo desenvolvido para realização de testes e as propostas desenvolvidas em pesquisa prévias. Observa-se que a presente pesquisa é uma continuação do trabalho apresentado na monografia de graduação de Rosa [13] e continuado na monografia de graduação de Santos e Moraes [30].

4.1 Propostas em estudos anteriores

O trabalho apresentado por Rosa em [13] têm como cerne o desenvolvimento de um Ajuste Dinâmico de Dificuldade híbrido, incluindo dados de desempenho e afetivos, para jogos de plataforma. Anteriormente, poucos estudos consideravam o uso de sensores em tempo real e apenas dois [43][89] comparavam as abordagens de ADD por desempenho e afetiva com o modelo híbrido. Todavia, se restringiam ao jogo *arcade Asteroids*. Após essa investigação, duas etapas foram definidas para o estudo:

1. Estimção da dificuldade, averiguando a eficiência de métodos computacionais para esse propósito e correlacionando-os estatisticamente à dificuldade experienciada pelos usuários do jogo; e
2. Comparação de modelos de ADD, verificando se o mecanismo híbrido mantém o jogador em um estado de fluxo e se possui um resultado melhor na dificuldade esperada pelo usuário. Foram considerados os seguintes casos para comparação:
 - Sem aplicação de ADD;
 - Aplicação de ADD de desempenho;
 - Aplicação de ADD afetivo; e
 - Aplicação de ADD híbrido.

No trabalho de Rosa, as adaptações eram feitas nas distâncias verticais e horizontais de componentes dos níveis, como plataformas e buracos. No trabalho que deu continuidade a este [30], desenvolvido por Santos e Moraes, foi analisado se a adição da variável de amplitude do salto da personagem afetaria o ADD quanto a sua eficácia em adequar a dificuldade e manter o jogador em um estado de fluxo. Mais especificamente, as alterações citadas foram realizadas somente no algoritmo do ADD de desempenho, devido à situação de pandemia de COVID-19 no momento em que a pesquisa foi desenvolvida. Esse contexto dificultou o acompanhamento e a organização de testes presenciais para captar os dados fisiológicos necessários para o ADD afetivo.

Nesse sentido, foi verificado se o ADD de salto estava devidamente implementado, sem interferir com os cálculos do ADD de desempenho do trabalho de Rosa. Para manter a consistência entre os trabalhos e ser uma comparação válida, a coleta de dados e momento de aplicação do ADD foram iguais ao trabalho de Rosa. Ademais, os casos de adaptação analisados foram:

- Sem aplicação de ADD;
- Aplicação de ADD de desempenho para adaptação de plataformas;
- Aplicação de ADD de desempenho para adaptação da amplitude do salto; e
- Aplicação de ADD de desempenho combinando adaptação de plataformas e da amplitude do salto.

4.2 Medição de dificuldade

Considerando o método de Ajuste Dinâmico de Dificuldade, foi proposta uma versão do método de Mourato [1][12] para avaliação da dificuldade em um nível de um jogo do gênero plataforma, descrito na Subseção 3.5.2. Mourato avalia a geração de níveis de jogos de plataforma e a adaptação para gerar estruturas mais complexas e não-lineares, como portões e alavancas. Nesse contexto, executa uma Geração de Conteúdo Procedural associada à variação da complexidade do ambiente, em vez de um ajuste da dificuldade dos componentes do jogo.

Em relação à representação de dificuldade, segue a ideia de Aponte [62] de que há relação entre as probabilidades de sucesso e fracasso durante a execução de tarefas, seja quando analisadas independentemente ou em um nível inteiro. Dessa forma, a abordagem é baseada em dois tipos de análise de nível:

1. O nível é decomposto em segmentos que representam partes independentes e são identificadas possíveis transições entre segmentos.

2. Cada componente é analisado, baseado na probabilidade de sucesso individual.

Para calcular a dificuldade do desafio como um todo, como ilustrado na Figura 4.1, é necessário dividi-lo em partes, mapeadas nas situações abaixo:

- a) Caminho reto sem possibilidade de falha.
- b) Um desafio de teste único com uma certa dificuldade, com chance de falha e passível de morte.
- c) Um desafio com penalidade mas que pode ser repetido:
 - (1) Com chance de falha, mas passível de repetição; e
 - (2) Com a possibilidade de um desafio adicional associado e chance de falha e passível de repetição e/ou morte.

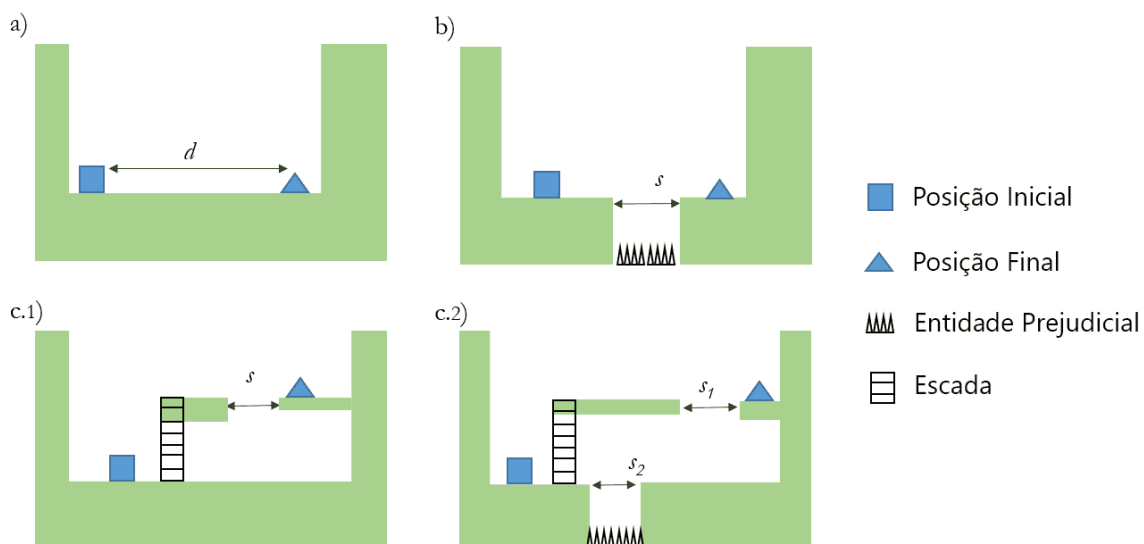


Figura 4.1: Representações de desafios em jogos de plataforma (adaptado de [12]).

Não é uma tarefa trivial identificar probabilidades para transições de segmento em um nível por jogos poderem ter diferentes movimentações e componentes. Assim, propõe-se observar características gerais e analisá-las em relação ao espaço e ao tempo.

Para a dificuldade espacial, considera-se o salto entre pontos, definindo o mínimo para ser performado. O cálculo é feito como o lançamento de um projétil do primeiro ponto a um segundo e medindo a possível margem de erro da sua trajetória até o ponto de destino, utilizando a interseção.

Por outro lado, para a dificuldade temporal, considera-se o período da entidade desafiadora, a janela de prejuízo e o tempo requerido para superar a entidade. Uma falha ocorre

se o jogador entra em contato com a entidade na janela de prejuízo ou se o movimento é executado tardiamente dentro do intervalo para superação do desafio.

Por fim, uma dificuldade espaço-temporal é mensurada pela multiplicação das probabilidades de sucesso dos casos anteriores. São caracterizadas por entidades dinâmicas, assim como plataformas móveis.

Mourato [1] mapeou níveis dos jogos *Super Mario Bros.* e *Little Big Planet* no formato de representação proposto e obteve sucesso testando a medição de dificuldade ao comparar a probabilidade de sucesso estimada e os resultados de jogadores de um grupo de amostra.

Considerando esse escopo, o trabalho de Rosa [13] utilizou desse método para estimação da dificuldade por meio dos componentes, incluindo alterações do método para considerar: (1) a influência do eixo horizontal na função de tempo da posição no eixo vertical; (2) a opção de novos tipos de componentes, como plataforma atravessável e buraco como uma entidade independente; e (3) as lógicas de sucesso do salto e de como o ADD será executado.

4.2.1 Implementação da estimativa da dificuldade

A medição da dificuldade de Rosa foi baseada no sucesso para executar saltos. Com isso, foram consideradas como triviais as ações de se movimentar e interagir com componentes que não causam dano, enquanto a probabilidade de sucesso entre plataformas e buracos determinará a dificuldade relativa a componentes individuais e ao nível.

O somatório da dificuldade de componentes específicos é feito para determinar a dificuldade de um nível. Os componentes analisados possuem salto como parte integral de seu desafio, sendo esses: buraco, plataforma padrão, plataforma móvel e plataforma atravessável. Para cada componente, é feita a identificação do ponto mais próximo de acordo com os tipos de plataforma a partir da origem do salto, definidos na Subseção 4.4.1.

O cálculo é feito como o lançamento de um projétil do ponto de origem P_0 ao ponto de destino P_1 e medindo a possível margem de erro da sua trajetória até o ponto de destino (P_2), utilizando a interseção. O ponto P_0 é determinado por um algoritmo de Busca em Largura (BFS), limitado pela direção de origem do jogador, encontrando o menor caminho a partir de um ponto de origem em termos de número de arestas no caminho. O ponto P_1 é o primeiro ponto de contato possível para o jogador, dependendo da direção de origem.

Plataformas móveis utilizam das mesmas considerações, considerando o caminho percorrido como posições válidas. Enquanto isso, para plataformas que podem ser atravessadas, a busca segue o mesmo padrão com o adicional de procurar na área diretamente abaixo delas.

Considera-se a trajetória de um projétil em função do tempo para representar o avatar e sua movimentação espacialmente. Assim, tendo x_0 como a velocidade inicial, v_0 a

velocidade inicial no eixo correspondente, y a posição atual no eixo vertical e $x(t)$ e $y(t)$ como as funções no tempo para os eixos horizontal e vertical, respectivamente, cumpri-se as Equações:

$$x(t) = x_0 + v_{0_x} * t \quad (4.1)$$

$$y(t) = y_0 + v_{0_y} * t + \frac{a}{2} * t^2 \quad (4.2)$$

No estudos, a aceleração e a desaceleração no eixo horizontal foram desconsideradas e a velocidade horizontal foi tratada como constante. Com isso e analisando g como a aceleração gravitacional no contexto do jogo, utiliza-se para o eixo vertical:

$$y(t) = y_0 + v_{0_y} * t - \frac{g}{2} * t^2 \quad (4.3)$$

Para os cálculos, o primeiro ponto é visto como a origem, não há um ângulo de lançamento e a velocidade constante é a máxima alcançável, assim tendo uma função quadrática interseccionando a origem, descrita pela Equação 4.4. Sendo v_{0_x} a velocidade máxima do eixo horizontal, v_{0_y} a velocidade do salto, x a posição no eixo horizontal e P_0 o ponto de origem, com $(x_0, y_0) = (0, 0)$.

$$f(x) = \frac{v_{0_y}}{v_{0_x}} * x - \frac{g}{2 * v_{0_x}^2} * x^2 \quad (4.4)$$

Ao interseccionar o projétil com duas linhas paralelas ao eixo cartesiano e que também interseccionam o ponto de destino, estima-se o possível desvio da trajetória que ainda é um salto válido, chamado de margem de erro, ilustrada na Figura 4.2 e descrita para o eixo x na Equação 4.5 e para o eixo y na Equação 4.6. Identifica-se dois valores: a altura do jogador quando alcança a plataforma em relação à origem (Δy) e a amplitude horizontal do salto (Δx). Assim, há um erro horizontal (m_x) e outro vertical (m_y), com $P_i = (x_i, y_i)$, ini_x e ini_y sendo as posições iniciais do ponto de destino, $ini_x = x_1$ e $ini_y = y_2 = y_1$. Nota-se que a plataforma mantém a mesma altura por toda sua extensão, caso contrário, é tratada como plataformas distintas acopladas.

$$m_x = \Delta x - ini_x = \Delta x - x_1 \quad (4.5)$$

$$m_y = \Delta y - ini_y = \Delta y - y_2 \quad (4.6)$$

Um erro menor ou igual à zero define uma destino não alcançável. Para valores positivos, a tolerância horizontal é calculada pelo valor de x para $f(x)$ igual a y_1 , obtido

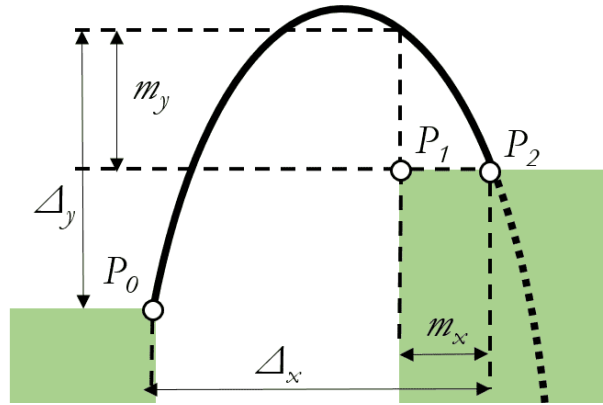


Figura 4.2: Representação gráfica do conceito de margem de erro [12].

por uma Equação de segundo grau e selecionando a raiz adequada. Assim, as seguintes fórmulas são obtidas para o eixo vertical de P_1 (y_1) e a vertical de P_2 (x_2), necessárias para calcular as margens de erro:

$$y_1 = f(x_1) = \frac{v_{0y}}{v_{0x}} * x_1 - \frac{g}{2 * v_{0x}^2} * x_1^2 \quad (4.7)$$

$$x_2 = (v_{0y} \pm \sqrt{v_{0y}^2 - 2 * g * y_2}) * \frac{v_{0x}}{g} \quad (4.8)$$

As margens de erro são valores relativos à distância total ao longo do eixo, precisando normalizá-los com distância total no eixo horizontal (x_r) e o ponto máximo no eixo vertical (y_v) e seguindo as Equações para margem de erro horizontal (M_x), margem de erro vertical (M_y) e margem de erro espacial final (M_e):

$$x_r = \frac{2 * v_{0y} * v_{0x}}{g} \quad (4.9)$$

$$y_v = \frac{v_{0y}^2}{2 * g} \quad (4.10)$$

$$M_x = \frac{m_x}{x_r} \quad (4.11)$$

$$M_y = \frac{m_y}{y_v} \quad (4.12)$$

$$M_e = M_x * M_y \quad (4.13)$$

Conforme indicado por Mourato [96], o salto pode ser performado antes no eixo horizontal para alcançar uma determinada altura. Para isso, calcula-se a distância mínima

(c_{min}) e a máxima (c_{max}) de correção no eixo horizontal e utiliza-se a média entre elas, como visto na Figura 4.3. Nota-se que plataformas que possibilitam ser atravessadas não precisam empregar esse mérito. Considerando $limite_{min}(x)$ como o limite inicial da plataforma no eixo horizontal e $limite_{max}(x)$ como o limite final da plataforma no eixo horizontal, obtém-se:

$$c_{min} = (limite_{min}(x) < x_1) ? (x_1 - limite_{min}(x)) : 0 \quad (4.14)$$

$$c_{max} = ((limite_{min}(x) < x_2) \wedge (limite_{max}(x) < x_2)) ? (x_2 - limite_{min}(x)) : (limite_{max}(x) - limite_{min}(x)) \quad (4.15)$$

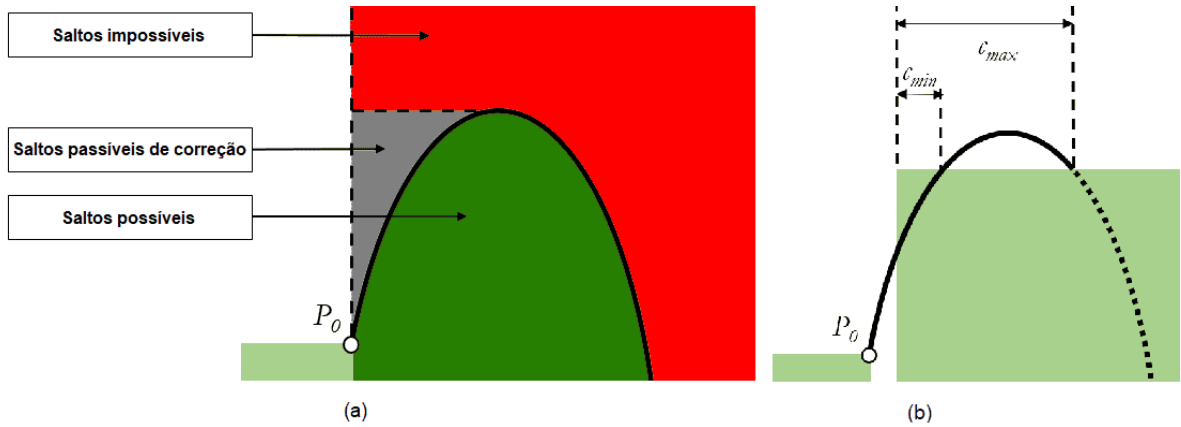


Figura 4.3: (a) Classificação do salto de acordo com a posição de destino (P_1) em relação à origem (P_0); (b) Valores de correção máximo e mínimo para tornar possível o salto de uma dada trajetória (adaptado de [12]).

No caso da plataforma móvel, que possui os méritos espacial e temporal, ocorre a multiplicação das probabilidades de erro. Para a dificuldade espacial, determina-se a dificuldade média entre os casos de sucesso pelos quais a plataforma percorre em algum momento. Para a dificuldade temporal, determina-se o período de tempo requerido para realizar o salto, obtido pela associação da velocidade da plataforma e do espaço em que não é possível saltar com sucesso, ou seja, as posições capazes de saltar dividido pela soma de todas as posições.

4.3 Dados afetivos

O sensor de captura da Atividade Eletrodérmica (EDA) é utilizado no estudo de Rosa [13], verificando a condutância da pele e fornecendo uma base para identificação dos níveis de

excitação, tensão e frustração. Para isso, a pulseira *Empatica E4*¹ foi utilizada. A pulseira *E4* é um dispositivo de pesquisa vestível que oferece aquisição de dados fisiológicos em tempo real e ferramentas para análise e visualização.

A pulseira possibilita uma monitoração discreta, acurácia e precisão nos dados captados, acesso aos dados e desenvolvimento de aplicativos próprios. Utilizou-se as funções de relógio interno, marcação de eventos e EDA para o estudo. Em suma, a pulseira fornece uma maneira de capturar a condutividade elétrica por meio da pele. Para isso, passa uma quantidade minúscula de corrente entre dois eletrodos em contato com a pele.

Os dados da pulseira já vem filtrados por serem otimizados para uso em tempo real e em condições adversas. Por outro lado, alguns problemas foram evidenciados previamente aos testes: pessoas com temperatura corporal muito baixa ou sob uso de certas medicações que suprimem as respostas do sistema nervoso simpático não conseguem ter seus dados captados pelos eletrodos EDA.

A *Empatica Inc.* fornece aplicativos para *Android* e *iOS* projetados para permitir o fluxo de dados em tempo real a partir dos dispositivos *E4* e de uma conexão *Bluetooth* desses com o celular. Todavia, a versão de desenvolvedor permite apenas a captação dos dados brutos. Com isso, o aplicativo foi modificado para enviar os dados brutos para um servidor de banco de dados por uma rede interna e o jogo foi adaptado para requisitar os dados recentes de forma ordenada.

Dentro do jogo, os dados brutos da EDA foram processados para geração de dois dados derivados: nível tônico da condutância da pele (mudanças lentas subjacentes) e resposta fásica de condutância da pele (picos que mudam rapidamente) [101], exemplificados na Figura 4.4. Por fim, foi adaptado um código aberto² que tinha como objetivo principal a detecção da excitação em tempo real usando recursos da Atividade Eletrodérmica. A adaptação consistiu em associar o código com os dados do sensor utilizado e encaminhar as análises para o controle do jogo. Com isso, obteve-se os níveis tônico, fásico e de excitação.

4.4 ADD híbrido

O jogo de Rosa possuía Ajuste Dinâmico de Dificuldade (ADD) (i) de acordo com fatores do desempenho do jogador, (ii) pela avaliação do estado afetivo relativo a Atividade Eletrodérmica (EDA) e (iii) um híbrido das opções anteriores. Nos três modelos descritos a dificuldade era adaptada pela variação horizontal e vertical de plataformas e buracos

¹<https://www.empatica.com/en-int/research/e4/>

²<https://github.com/ddessy/RealTimeArousalDetectionUsingGSR>

³Ilustração adaptada de <https://imotions.com/blog/galvanic-skin-response/>

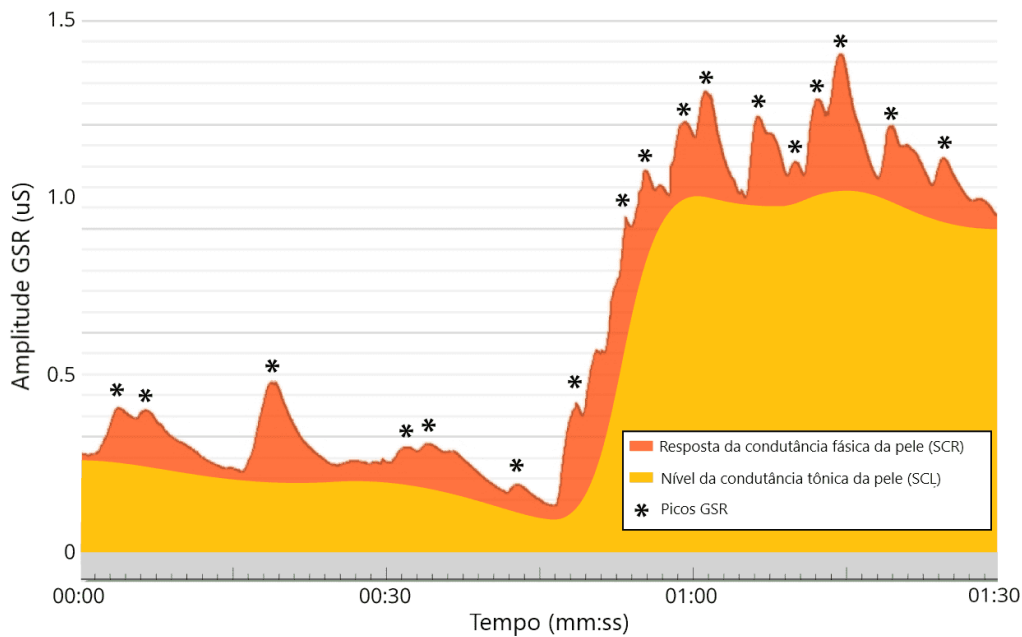


Figura 4.4: Exemplo da variação da amplitude GSR ao longo do tempo³.

a partir de variáveis de dificuldade globais que representavam o desempenho e o estado afetivo do jogador. A dificuldade para o modelo híbrido correspondia à média dos demais.

Os dados e processamentos para definição das dificuldades globais foram escolhidos a partir de estudos anteriores, sendo [7] e [102] para os componentes, e [12] e [100] para o desempenho do jogador. Com a primeira bateria de testes e a análise de seus resultados, essas escolhas foram verificadas, buscando analisar o que influenciava o desafio e o desempenho do usuário.

A implementação dos três modelos de ADD seguiu um padrão cíclico com 4 sistemas interligados, baseado nos estudos [4] e [5]. Os sistemas referem-se aos dados brutos do Jogador, o Monitoramento dessas métricas, a Análise correspondente e o Controle dos componentes que serão ajustados. A Figura 4.5 apresenta o diagrama dos sistemas utilizados para aplicação do ADD híbrido e um resumo das etapas associadas.

As variáveis do Jogador são captadas pelo Sistema de Monitoramento, dividindo entre os dados brutos do jogo e as medidas fisiológicas, as quais são obtidas por um sensor que examina a resposta afetiva do jogador. Após isso, os dados são filtrados, permitindo detectar e interpretar o desempenho e a emoção correspondentes às variáveis apresentadas no Sistema de Análise. Esses dados podem ser utilizados separadamente no modelo por desempenho e afetivo, respectivamente, ou em conjunto para o modelo híbrido. Após feita a análise, o resultado determina o que será adaptado no jogo pelo Sistema de Controle.

Para o caso afetivo, é usada uma proporção do nível fásico e do nível tônico medidos. A cada período especificado de tempo (2 segundos), a proporção é recalculada e a variação

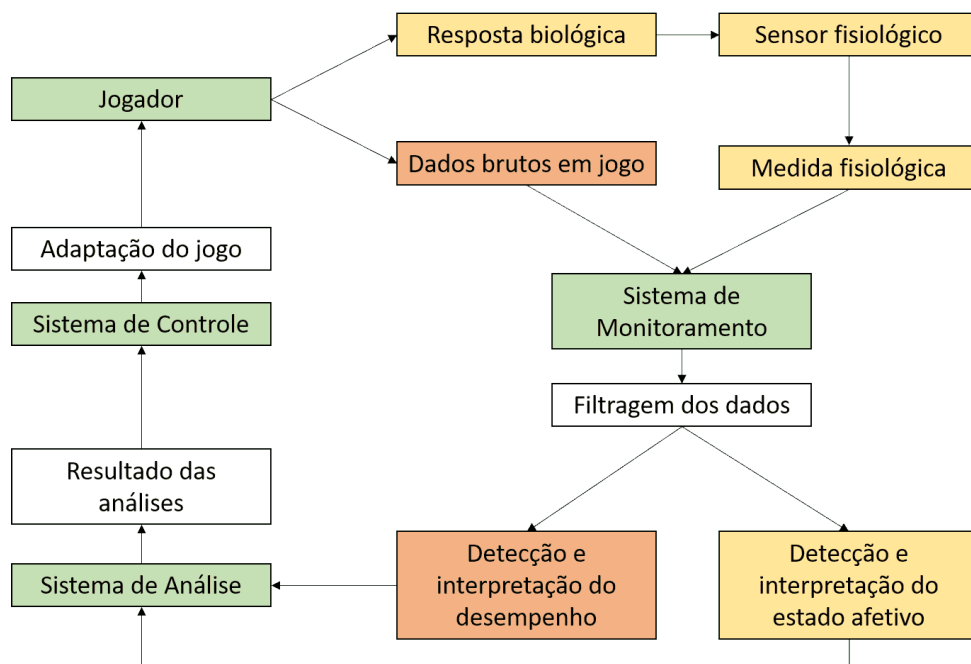


Figura 4.5: Diagrama cíclico do Ajuste Dinâmico de Dificuldade híbrido orientado pelo sistema. Os sistemas estão em verde, enquanto as etapas relacionadas ao desempenho do jogador estão em vermelho e à afetividade, em amarelo.

dela é a adicionada à dificuldade global.

Em relação ao desempenho, quando a morte é relacionada a uma queda, um contador é incrementado e a posição da morte é salva. No caso da morte ser em uma posição prévia ou igual a da morte anterior, a dificuldade global diminui (facilitando) conforme a proporção entre a distância atual e a anterior. No caso inverso, a dificuldade global aumenta (dificultado) em menor escala.

Quanto mais segmentos são completados sem a ocorrência de morte, a dificuldade global cresce de modo exponencial e, quando há morte, decresce exponencialmente. Essa lógica é aplicada também entre níveis e é a que causa maior variação no cálculo da dificuldade global. Ademais, é feita uma proporção e normalização entre o número de saltos de sucesso e falhos. Com isso, dificulta quando há uma maior incidência de sucesso e facilita quando aumenta o número de falhas.

O valor global da dificuldade é limitado de acordo com a progressão de níveis, conforme explicitado nas Equações 4.16 e 4.17. Considera-se que dif é a dificuldade global (variando entre 0 e 1 em todos os casos), $maxDifNivel$ é a dificuldade limite do nível, $nivel$ é o nível atual e $qtdNiveis$ é a quantidade de níveis. Por exemplo, no caso de haver 12 níveis, o primeiro pode variar entre 0 e 0,83, o sexto entre 0 e 1, o penúltimo entre 0,16 e 1 e o último entre 0,25 e 1. Enquanto isso, no caso de 24 níveis, o primeiro varia entre 0 e 0,79 e o penúltimo entre 0,20 e 1.

$$maxDifNivel = nivel/qtdNiveis \quad (4.16)$$

$$dif = (dif > maxDifNivel + 0,75) ? (maxDifNivel + 0,75) : ((dif < maxDifNivel - 0,75) ? (maxDifNivel - 0,75) : dif) \quad (4.17)$$

As adaptações do ADD foram programadas para serem feitas em 3 possíveis momentos:

1. Ao criar segmentos e níveis:

- Na geração de cada segmento, utiliza o valor da dificuldade global atual para modificação dos componentes.

2. Ao morrer:

- Guarda um valor de dificuldade global ao morrer e o utiliza para modificar o segmento atual e os próximos.

3. Em tempo real:

- Verifica se o mapa deve ser alterado a cada variação da dificuldade global.

A adaptação do nível é igual para todas as opções acima, diferindo apenas em relação a periodicidade da aplicação. Ou seja, a forma que os componentes são modificados não diverge, mas o momento que isso é feito pode variar. Além disso, podem ser executadas em conjunto, no caso de adaptar tanto ao morrer quanto ao criar novos segmentos e níveis.

4.4.1 Extração de dados

A extração de dados provenientes do jogo e de um sensor da Atividade Eletrodérmica (EDA) foi usada para medir, respectivamente, o desempenho e o estado afetivo do jogador. Os dados coletados na estrutura dos estudos são:

- Temporal:
 - Tempo para completar segmento/nível;
 - Tempo em cada vida;
 - Tempo andando para esquerda, direita ou parado; e
 - Tempo em cada estado do jogador (agachar, saltar, atacar, atirar, dentre outros).

- Quantitativo:
 - Quantidade de entradas em um determinado estado do avatar:
 - * Contabilização de sucessos e falhas de ataques ao acertar inimigos ou obstáculos; e
 - * Contabilização de sucessos, falhas e ações randômicas de saltos.
 - Quantidade de interações com cada componente do mapa;
 - Quantidade de colecionáveis coletados; e
 - Quantidade de mortes.
- Informações específicas:
 - Causa da morte, como armadilha, inimigo ou queda (em abismo ou ácido); e
 - Distância percorrida até uma morte.

Para cada dado também são feitas proporções com valores prévios e em relação ao total de um segmento, de um nível ou de uma vida. Para os méritos desse estudo, uma única interação com os componentes que causam morte já é fatal, em vez do acúmulo de múltiplos danos para isso ocorrer.

Sucesso do salto

A determinação da estatística de sucesso do salto em um jogo de plataforma possui uma lógica específica por não ser uma medida uniformizada ou pré-estabelecida. Para compreender como é essa mensuração, foi preciso determinar os objetivos do jogador:

- Objetivo principal: chegar ao final do nível; e
- Objetivo secundário: exploração do nível, obtendo colecionáveis e eliminando inimigos.

Com os objetivos do jogador estabelecidos, foram definidos os tipos de plataforma pela origem do salto, no caso, qual a origem esperada do jogador em relação à posição da plataforma. Assim, foram definidos os tipos Superior Direita, Superior Esquerda, Inferior Direita, Inferior Esquerda e Esquerda Direita, de acordo com as seguintes características:

- Superior se refere à plataforma de destino do salto estar em uma posição mais alta que a plataforma de origem;
- Inferior se refere à plataforma de destino do salto estar em uma posição mais baixa que a plataforma de origem;

- Direita se refere à plataforma de destino do salto vir depois horizontalmente que a plataforma de origem;
- Esquerda se refere à plataforma de destino do salto vir antes horizontalmente que a plataforma de origem;
- Esquerda Direita se refere a buracos que a origem e o destino estão em uma mesma posição vertical.

Os tipos de plataforma estão exemplificados na Figura 4.6. Observa-se que a especificação do tipo de plataforma é definida pelo arquivo de imagem para criação do nível e que essas definições são utilizadas para modificação dos componentes, mantendo-os em um padrão de *design* definido.

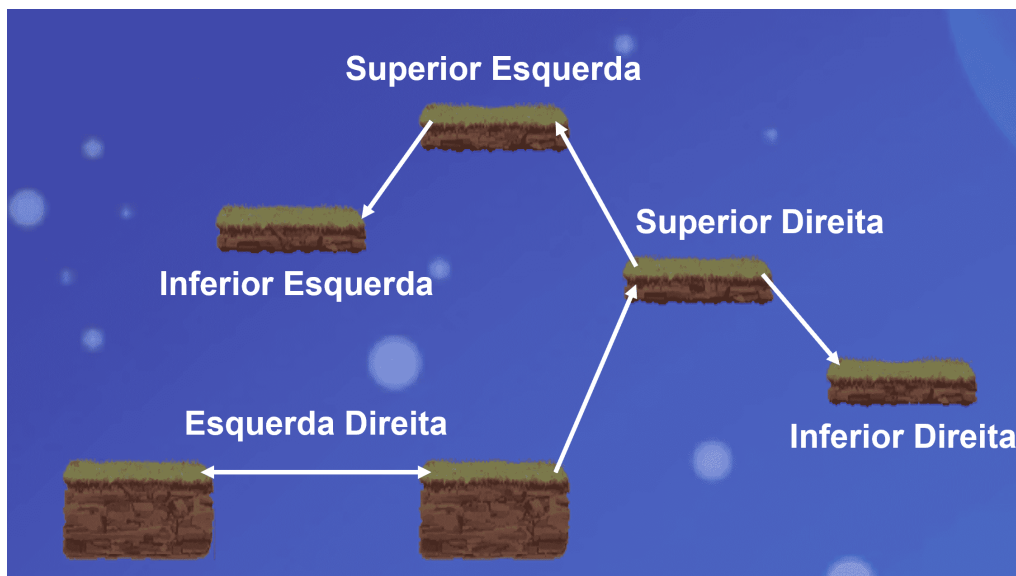


Figura 4.6: Representações dos tipos de plataforma no jogo. As setas indicam a origem e o destino do salto, com a plataforma correspondente possuindo sua nomenclatura associada.

Considerando as determinações especificadas, definiu-se uma ordem de condições para categorizar um salto como de sucesso, falho ou randômico:

1. Falha ao morrer durante salto.
2. Ao colidir com plataforma diferente da origem:
 - (a) Se o avatar colidir alguma parte de seu corpo com alguma plataforma antes dos pés:
 - i. Sucesso se, mesmo tendo colidido no caminho, a plataforma final for abaixo da inicial, quando do tipo Inferior;

- ii. Falha se a plataforma final não for a mesma de início, a plataforma final for abaixo da inicial, quando do tipo Inferior:
 - Caso em que tenta saltar de uma plataforma para outra, porém cai para outra inferior.
 - iii. Sucesso se os pés entraram em contato com outra plataforma ao final do salto; e
 - iv. Falha se a plataforma final é a mesma de início e a distância no eixo vertical entre as posições de início e final do salto for menor do que um ladrilho:
 - Caso em que tenta saltar para uma plataforma, porém não a alcança com os pés, apenas com outra parte superior do corpo.
- (b) Sucesso se alcançar uma parte da mesma plataforma com diferença vertical maior que o espaço de um ladrilho:
- Caso de plataforma que possui diferença de altura em si, a exemplo de uma escada.
- (c) Sucesso se ultrapassou um buraco:
3. Sucesso se obteve colecionável ou acertou inimigo.
4. Randômico se não encaixa em nas opções acima.

A partir desse tipo de dado, é possível mensurar o desempenho do jogador em saltos enquanto percorre um nível, com base a como os saltos são realizados e as consequentes interações com plataformas, buracos, colecionáveis e condições de morte.

4.4.2 Análise de dados de desempenho

O ADD de desempenho pode considerar diversos parâmetros para o Sistema de Análise, seguindo como base os dados coletados. Nesse contexto, o processamento desses dados pode ser feito de acordo com a metodologia aplicada para o jogo. Assim, foram definidas heurísticas para calcular o fator de desempenho do usuário, o qual é um valor entre 0 e 1 aplicado sobre a dificuldade global, calculado a partir dos seguintes parâmetros de entrada:

- Quantidade de mortes (em adaptação de plataformas):
 - Ao criar segmento de nível, facilita se ocorreu morte nos segmentos anteriores e dificulta se não ocorreu morte nos segmentos anteriores:
 - * Ao facilitar,

$$(1 - 1.1^{-qtdMortes})/10 \quad (4.18)$$

* Ao dificultar,

$$0,125 - (0,125/qtdMortesSeg) \quad (4.19)$$

* Considera-se $qtdMortes$ como a quantidade de mortes e $qtdMortesSeg$ como a quantidade de segmentos sem morte.

• Distância percorrida até morrer (em adaptação de plataformas):

– Ao morrer, facilita se posição da morte atual for menor do que da morte anterior e dificulta se a posição da morte atual for maior do que a morte anterior:

* Ao facilitar, diminui

$$(1 - (posMorte/posMorteAnt))/50 \quad (4.20)$$

* Ao dificultar, diminui

$$(1 - (posMorteAnt/posMorte))/100 \quad (4.21)$$

* Considera-se $posMorte$ como a posição atual de morte e $posMorteAnt$ como a posição da morte anterior.

• Quantidade de saltos de sucesso:

– Ao morrer e ao criar segmento do nível (em adaptação de plataformas), facilita se ocorreram 5 vezes mais saltos falhos do que de sucesso e dificulta se ocorreram 5 vezes mais saltos de sucesso do que falhos:

* Ao facilitar, diminui $propSalto$

* Ao dificultar, aumenta $propSalto$

* Considera-se

$$propSalto = \frac{(saltoFalho \leq 0) ? saltoSucesso : \frac{saltoSucesso}{saltoFalho}}{saltoSucesso + saltoFalho} \quad (4.22)$$

dividido por valor constante, o valor constante de 100 ao morrer e 50 ao criar segmento do nível, $saltoFalho$ como a quantidade de saltos falhos e $saltoSucesso$ como a quantidade de saltos de sucesso durante o nível.

– Ao morrer e ao criar segmento do nível (em adaptação do salto), facilita se ocorreram mais saltos falhos do que de sucesso e dificulta se ocorreram mais saltos de sucesso do que falhos:

* Ao facilitar, diminui valor constante.

* Ao dificultar, aumenta

$$(\text{saltoFalho} \leq 0) ? \text{saltoSucesso} : \frac{\text{saltoSucesso}}{\text{saltoFalho}} \quad (4.23)$$

dividido por valor constante.

* Considera-se o valor constante de 100 ao morrer e 50 ao criar segmento do nível, *saltoFalho* como a quantidade de saltos falhos e *saltoSucesso* como a quantidade de saltos de sucesso durante o nível.

O valor de dificuldade global começava com 0,25 nos testes contendo adaptação de plataformas e o valor de dificuldade global começava como 0,5 nos testes contendo adaptação de salto. No caso combinado, havia duas dificuldades globais, variando conforme suas heurísticas específicas.

4.4.3 Adaptações a partir da dificuldade

Um mapa dos componentes do nível é gerado quando este é iniciado, contendo a dificuldade total, a dificuldade total por componentes, sub-mapas para cada segmento e suas dificuldades correspondentes. Cada componente (organizado por tipo) possui uma dificuldade associada, a direção para cálculo da dificuldade pelos tipos vistos na Subseção 4.5.3, a origem e o destino do salto e as configurações de possíveis movimentações. A dificuldade global do jogador (*dif*) é comparada com a dificuldade média do segmento (*difSeg*). A diferença gera a dificuldade que deve ser modificada (*difModif*), seguindo a Equação 4.24 e variando entre -1 e 1 em relação à dificuldade atual, dependendo se deve diminuir (negativo) ou aumentar (positivo).

$$\text{difModif} = \text{dif} - \text{difSeg} \quad (4.24)$$

Caso o valor absoluto da dificuldade que deve ser modificada seja maior que o limiar do eixo x ou y, verifica-se quais os componentes específicos que devem ser alterados. Ou seja, a dificuldade para um eixo específico só é alterada se o absoluto dela for maior que o limiar correspondente (*limiar_x* e *limiar_y*), definidos nas Equações 4.25 e 4.26 de forma a não permitir saltos impossíveis.

$$\text{limiar}_x = 1/x_r \quad (4.25)$$

$$\text{limiar}_y = 1/y_v \quad (4.26)$$

Para que não haja a normalização da dificuldade de todos os componentes de um segmento e que se mantenha a variabilidade do nível, em cada componente é aplicado um padrão exponencial, no qual $diffComp$ é a dificuldade atual do componente (verificada sempre antes de alguma mudança), $auxDiffComp$ é a dificuldade temporária que auxilia no cálculo e $diffCompModif$ é a dificuldade que deve ser modificada no componente, conforme apresentado nas Equações:

$$auxDiffComp = diff - diffComp \quad (4.27)$$

$$diffCompModif = diffModif^{(1+diffModif-auxDiffComp)} \quad \forall diffModif > 0 \quad (4.28)$$

$$diffCompModif = diffModif^{(1-diffModif+auxDiffComp)} \quad \forall diffModif < 0 \quad (4.29)$$

Observa-se que cálculos referente à parábola são feitos assim que o nível começa e só precisariam ser refeitos no caso de configurações do componente ou valores de velocidade, aceleração ou gravidade do avatar alterarem. Caso o valor absoluto da dificuldade que deve ser modificada de cada componente seja maior que o limiar do eixo x ou y, esse será modificado (adicionando ou removendo) em uma determinada quantidade de ladrilhos (qtd_{tiles_x} e qtd_{tiles_y} , respectivamente):

$$qtd_{tiles_x} = \lfloor diffCompModif / limiar_x \rfloor \quad (4.30)$$

$$qtd_{tiles_y} = \lfloor diffCompModif / limiar_y \rfloor \quad (4.31)$$

As alterações modificam o tamanho e a posição dos buracos e das plataformas, adicionando (ao facilitar) ou removendo (ao dificultar) ladrilhos da região de destino do salto. Para as plataformas do tipo Superior, adiciona ou move para cima para dificultar e remove ou move para baixo para facilitar. O inverso se aplica às plataformas do tipo Inferior.

O mapeamento de posições ocupadas por componentes foi feito para auxiliar nesse processo de adaptação. Sendo que, ao final do processo, as dificuldades relativas a cada componente são atualizadas, assim como do tipo do segmento e do nível. Nas Figuras 4.7, 4.8 e 4.9 é demonstrado como o nível pode variar com a aplicação do ADD.

Por sua vez, o estudo de Santos e Moraes [30] testou apenas o ADD de desempenho e implementou a adaptação de configurações do avatar, alterando a amplitude do salto. Como no ADD de plataforma já existiam contadores de morte e de saltos de sucesso e

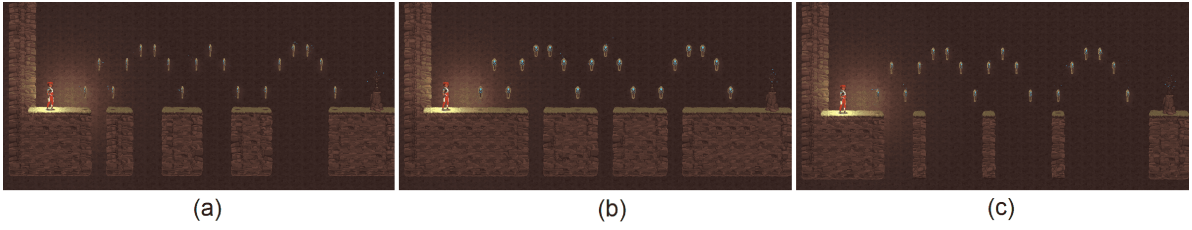


Figura 4.7: Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0,1 e (c) 0,9, respectivamente.

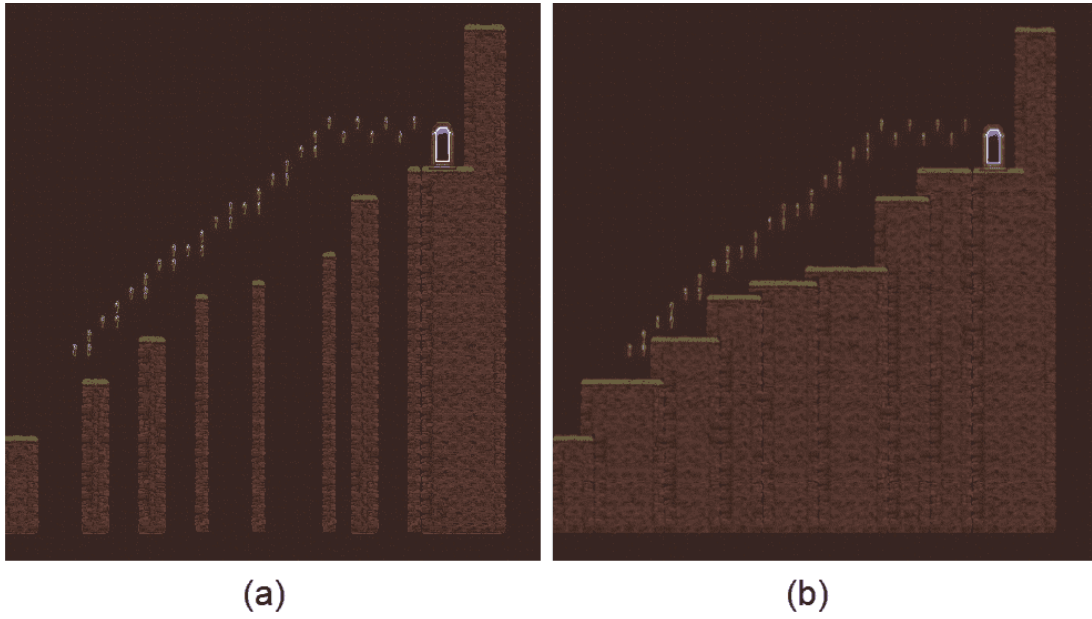


Figura 4.8: Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0, respectivamente.

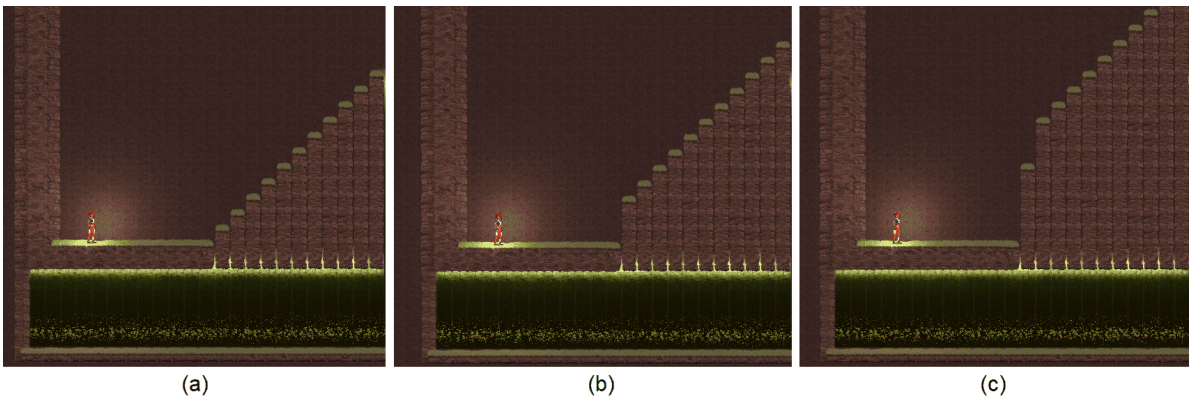


Figura 4.9: Captura de tela de segmentos demonstrando a diferença de níveis (a) sem a aplicação do ADD e com valor de dificuldade global (b) 0,7 e (c) 1, respectivamente.

falhos, as mesmas variáveis foram utilizadas como parâmetro para o Sistema de Análise dessa adaptação, facilitando ao aumentar a amplitude do salto e dificultando ao diminuir,

exemplificado na Figura 4.10.

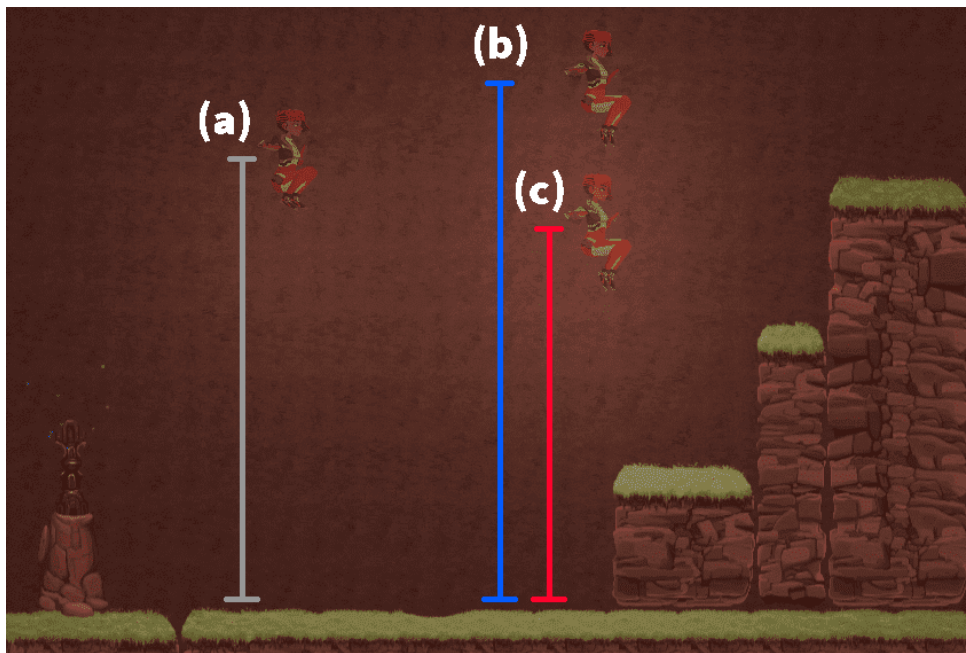


Figura 4.10: Captura de tela do mesmo segmento com as diferentes alturas máximas do salto. (a) Amplitude máxima sem ADD aplicado; (b) Com ADD de salto aplicado para o valor de dificuldade global mais fácil; e (c) Com ADD de salto aplicado para o valor de dificuldade global mais difícil.

4.5 O jogo

Considerando que o desenvolvimento de um jogo eletrônico não era propósito dos trabalhos de Rosa e de Aguiar e Fernandes, assim como do presente trabalho, os esforços se concentraram na adequação de um jogo já desenvolvido às necessidades dos estudos. Com isso, os seguintes requisitos foram definidos para escolha de um jogo:

- Licença para uso não comercial e código fonte aberto para que as alterações do jogo fossem possíveis;
- Estruturação dos componentes do jogo de modo que fosse possível modificar o código para avaliar e armazenar dados brutos da sessão de jogo, associar ao uso de um sensor afetivo e criar um sistema de Ajuste Dinâmico de Dificuldade;
- Presença de componentes comuns ao gênero de plataforma, baseado nos estudos de Minkkinen [41];
- Controles e interações simples para que jogadores menos experientes conseguissem se familiarizar com a jogabilidade sem ocasionar frustração;

- Jogo para um único usuário, em vista das pesquisas focarem nesse tipo de jogo e novas variáveis serem introduzidas em casos de múltiplos jogadores; e
- Execução em Sistema Operacional *Windows*, *Linux* e *macOS*, de acordo com os computadores que poderiam ser utilizados.

A partir dos requisitos e objetivos definidos, o jogo de plataforma *The Explorer: 2D* foi escolhido, ilustrado na Figura 4.11. Este jogo foi elaborado pela *Unity Technologies* e serve como em uma coleção de ferramentas, sistemas e componentes de jogabilidade feitos no motor de jogo *Unity* para que artistas e *designers* possam aprender a utilizá-lo. O jogo foi resumido pela equipe desenvolvedora da seguinte forma⁴: "Explore o antigo e misterioso planeta alienígena onde nossa principal engenheira, Ellen, caiu. Evite os perigos e derrote os inimigos que espreitam nas antigas ruínas desta civilização alienígena."



Figura 4.11: Captura de tela da versão original do jogo *The Explorer: 2D*.

4.5.1 Motor de jogo

A primeira versão do jogo data de dezembro de 2017 e foi implementada na plataforma *Unity* com suporte para versões a partir da 2017.3.0. O jogo recebia constantes atualizações para correções de erros e adaptações para novas versões do motor, com a atualização mais recente sendo de 9 de abril de 2021 para versão 2020.3 do *Unity*.

⁴<https://learn.unity.com/project/2d-game-kit>

O *Unity*⁵ é um motor de jogo para desenvolvimento de jogos 2D e 3D, criado pela *Unity Technologies*. Observa-se que um motor de jogo consiste em um programa de computador e/ou conjunto de bibliotecas que cria abstrações para o desenvolvimento de jogos eletrônicos. Um motor geralmente fornece funções de renderização gráfica e simulações físicas com suporte para animações, sons, gerência de memória e outras funções. No contexto dos trabalhos de Rosa [13] e Soares e Moraes [30], o projeto foi construído como aplicação para *desktops* do Sistema Operacional *Windows*.

O motor tem como objetivo simplificar o fluxo de desenvolvimento de jogos, contendo uma ferramenta de edição, permitindo o uso de linguagens de programação robustas e sem limitar comportamentos dos objetos aos módulos incorporados à aplicação. Assim como na estrutura original do jogo *The Explorer: 2D*, a codificação foi feita em *CSharp* (C#), uma linguagem de programação multiparadigma de tipagem forte.

4.5.2 Versão original do jogo

O jogo escolhido se assimila ao *Super Mario Bros.*, com o jogador controlando um avatar (Ellen) que pode se movimentar para frente ou para trás, pular e cair, almejando percorrer um caminho no qual irá superar obstáculos para alcançar o final do percurso. O jogador também pode agachar, executar um ataque corpo-a-corpo e atirar, com a câmera movendo conforme a posição do jogador e podendo ser elevada para cima ou para baixo pelos controles.

O avatar possui configurações definidas por meio dos seguintes parâmetros variáveis: velocidade, aceleração, desaceleração, proporção da velocidade ao empurrar um objeto, dano recebido e dano executado, número de vidas, e posição e velocidade da câmera. Esses fatores podem se relacionar ao eixo horizontal (ao andar) e ao vertical (ao saltar ou cair). Ademais, os controles básicos para as ações do avatar são: movimento lateral, salto, olhar para cima, agachar/olhar para baixo, ataque corpo-a-corpo e atirar. No menu principal, antes de iniciar o jogo, ou no menu de pausa, disponível durante a sessão de jogo, é permitido modificar o volume, ver os controles ou sair do jogo.

Há 5 níveis na versão original do jogo e novas podem ser desenvolvidas pela ferramenta de edição do *Unity*. Os níveis incluem inimigos, uma quantidade máxima de vidas, recomendo o nível caso chegue ao limite estabelecido, e caminhos não-lineares, necessitando executar pequenos quebra-cabeças, como encontrar chaves e painéis de pressão que abrem portas. A interface do jogo, apresentada na Figura 4.12, possui as informações visuais das vidas restantes e do inventário de chaves coletadas.

⁵<http://unity3d.com/pt>



Figura 4.12: Exemplificação da HUD na versão original do jogo, contendo vidas, inventário e interface de diálogo.

4.5.3 Modificações do jogo

O jogo *The Explorer: 2D* possui uma boa estruturação e múltiplas possibilidades de *design*, porém diversas alterações foram necessárias para adequá-lo à proposta dos trabalhos de [13]. Primeiramente, foi desenvolvida a geração de níveis a partir de imagens *bitmap*, como visto nas Figuras 4.13 e 4.14. Como a proposta original do jogo é voltada para *designers*, foi necessária um novo formato de formulação de níveis de modo mais ágil para desenvolvedores. Assim, os níveis originais do jogo foram descartados, pois o estudo não possui o conteúdo de histórias e missões. Ademais, esse modelo de geração organiza os componentes de modo padronizado para permitir que sejam adaptados durante o jogo.

No modelo de criação por imagem, cada pixel representa um componente no mapa em grades do jogo, sendo que a cor do pixel identifica qual o componente correspondente. Ou seja, o jogo foi reestruturado para funcionar em um padrão de grade (ladrilhos quadrados de tamanho fixo), em vez de posicionar sem uma referência numérica inteira. Para isso, como ilustrado na Figura 4.15, todos os componentes do jogo foram categorizados e implementados nesse formato:

- Personagem jogável (Ellen).
- Plataformas:
 - Buraco, posição em que pode ocorrer queda do avatar;
 - Plataforma Básica, posição estática utilizada para travessia do avatar;

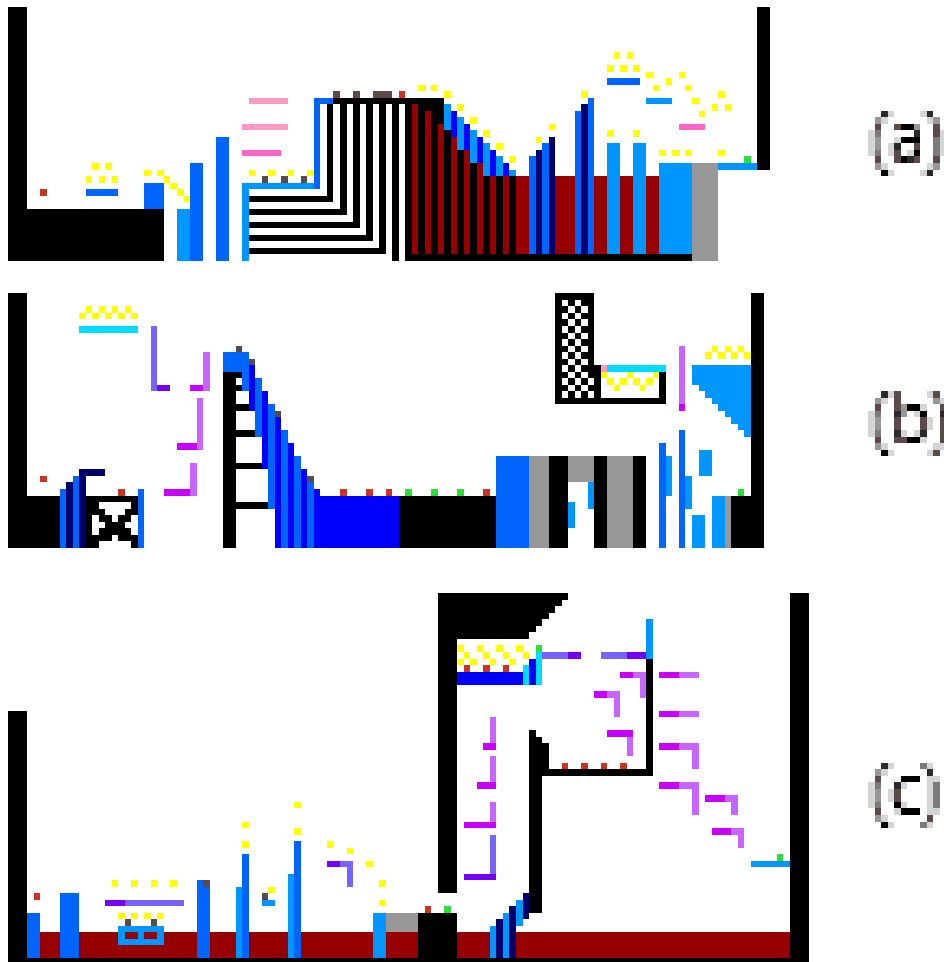


Figura 4.13: Arquivos de entrada para geração do nível.

- Plataforma Móvel, posição dinâmica, com movimento vertical e/ou horizontal, para travessia do avatar; e
- Plataforma Atravessável, posição estática ou dinâmica em que não há colisão ao passar pela parte inferior da plataforma.
- Elementos gerais:
 - Objeto destrutível, quebrando ao executar ataque corpo a corpo que o acerte:
 - * Coluna; e
 - * Parede.
 - Objeto capaz de ser empurrado ao se aproximar, objetivando alcançar posições mais altas ou resolver quebra-cabeça;
 - *Checkpoint*, o qual salva o progresso ao entrar em contato:

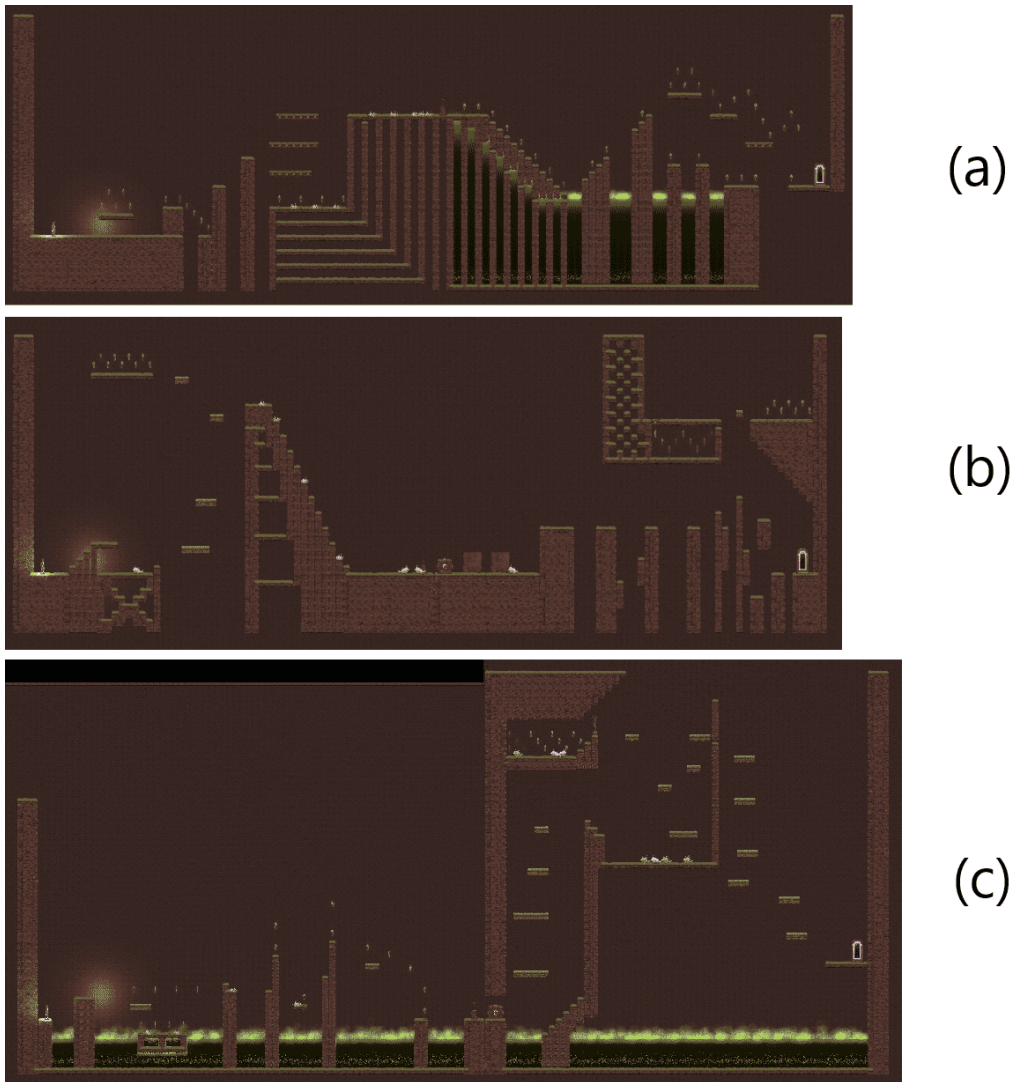


Figura 4.14: Níveis gerados a partir dos arquivos de entrada.

- * Área invisível, sem visual, geralmente para determinar a posição inicial do jogador no nível; e
- * Poste, interface visual que indica o progresso sendo salvo ao acender uma luz.
- Portal de saída, no qual o nível é finalizado ao entrar em contato.
- Elementos que causam morte:
 - Abismo, buracos em que o avatar morre ao cair, delimitado por toda a área na parte inferior do mapa ;
 - Ácido, região com aparência líquida e hostil; e
 - Espinho, um tipo de armadilha que não pode se aproximar.

- Colecionáveis:
 - Chave, o correspondente à moeda dos jogos *Super Mario Bros.* e *Sonic the Hedgehog*.
- Inimigos:
 - *Chomper*, executa ataques corpo a corpo enquanto se movimenta em direção ao jogador; e
 - *Spitter*, executa ataques à distância, atirando a partir de uma posição fixa.

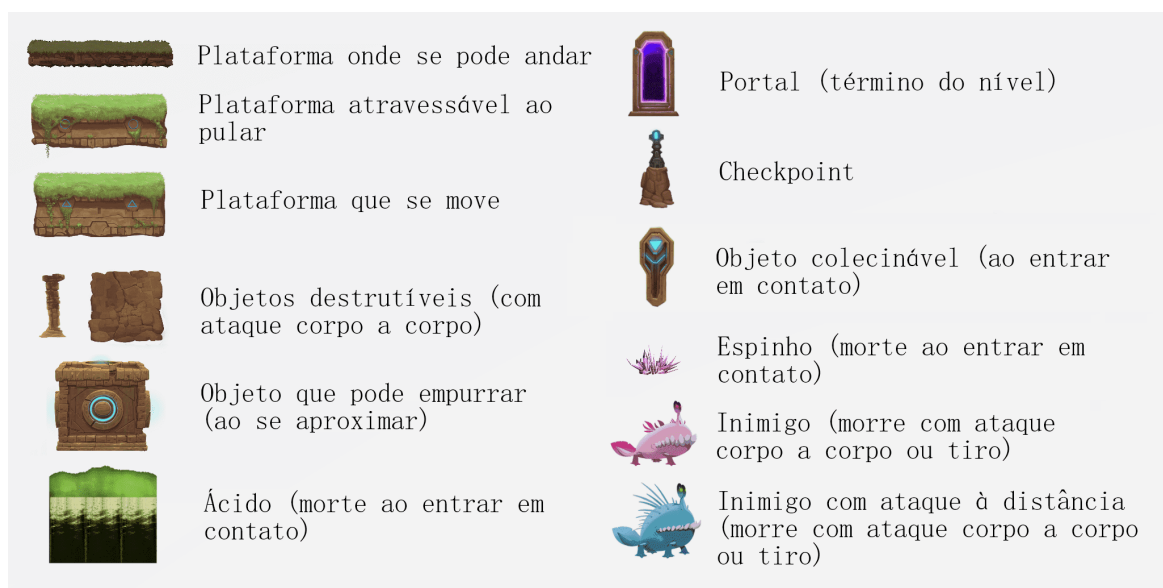


Figura 4.15: Ilustração dos principais componentes do jogo, conforme é apresentada ao usuário pelo menu inicial e pelo de pausa.

Na geração de níveis por imagem, é feita uma organização de objetos com todos os componentes do mapa divididos por segmento e tipo. Os componentes podem ser acessados para análise e adaptação, permitindo a a estimacão da dificuldade e o ajuste dinâmico. Além disso, diversos componentes tiveram que ser modificados para melhoria de performance, reajustes de versão e compatibilidade com as alterações da estrutura do jogo.

Com o foco da imersão, houve uma preocupação visual para manter coerência e harmonia no posicionamento dos componentes, especialmente quando estão próximos e são de um mesmo tipo. Para isso, segue regras visuais pré-determinadas para escolher entre os distintos modelos de ladrilho para construir plataformas, conforme mostrado na Figura 4.16.

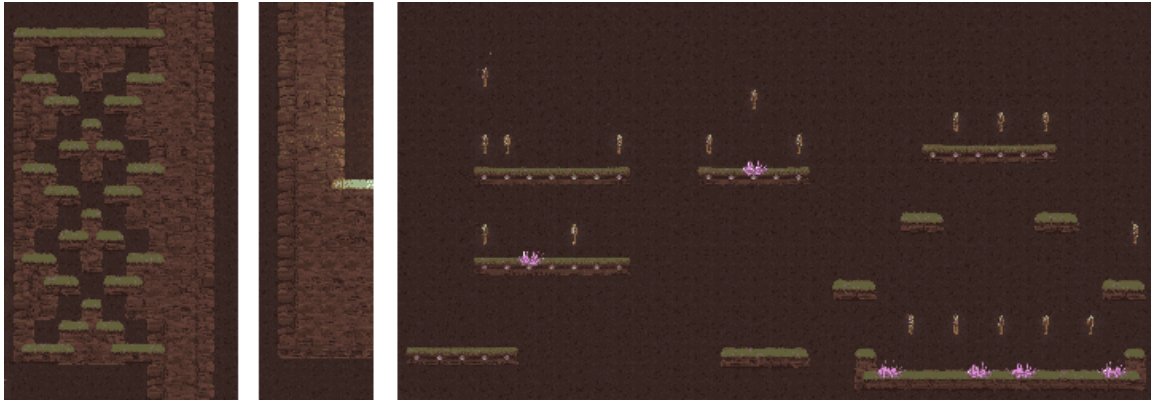


Figura 4.16: Exemplificação da modelagem de ladrilhos das plataformas.

Observa-se que, ao alcançar um portal de saída, inicia-se o nível seguinte, podendo aplicar um questionário entre os níveis. Caso não haja mais níveis para serem carregados, o jogo finaliza com um questionário e uma tela de agradecimento. Também foram adicionadas telas de apresentação contendo os objetivos, os componentes principais e os controles do jogo (também acessíveis no menu de pausa) e, antes de começar o primeiro nível, um questionário sociodemográfico, ilustrados na Figura 4.17. Essas interfaces foram desenvolvidas para agregar na imersão do usuário, não precisando sair do jogo para responder um questionário ou ter respostas visuais de progresso.

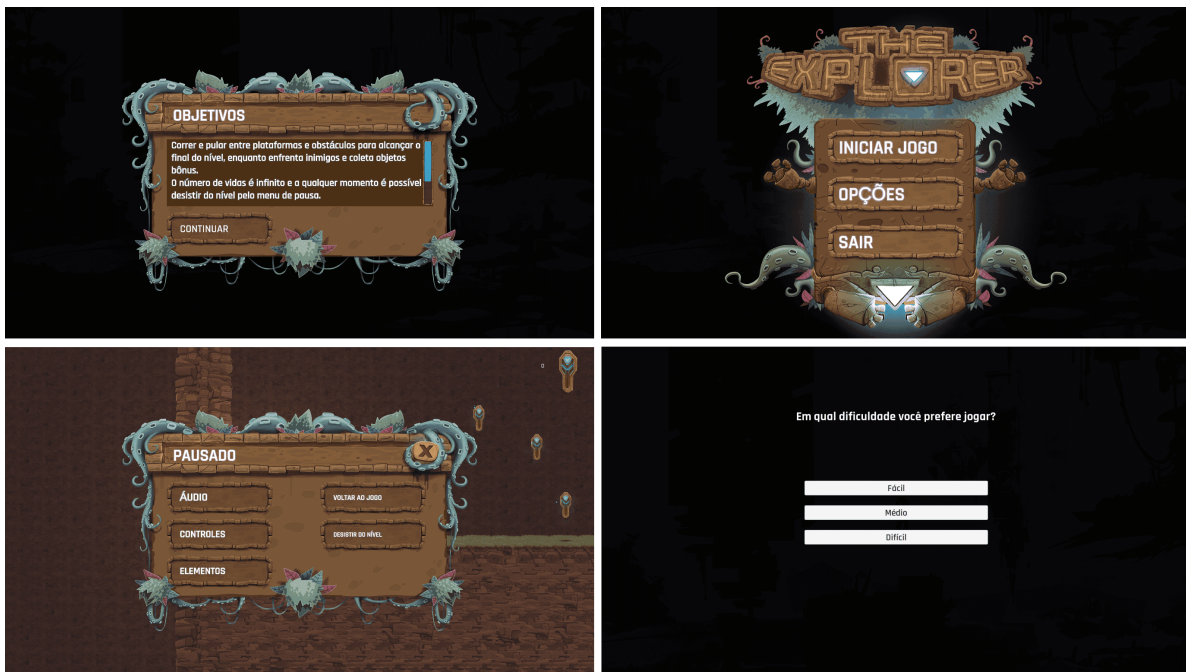


Figura 4.17: No sentido horário, interface dos objetivos, tela inicial, questionário e menu de pausa.

Os menus inicial e de pausa foram adaptados para incluir as novas opções de desistir

de um nível e de escolher entre modelos pré-definidos de entrada dos controles do avatar. Além disso, a HUD foi modificada para incluir a contagem de chaves coletadas e retirar o limite de vidas, pois o jogador possui vidas infinitas no contexto do estudo. Logo, o jogador pode completar ou desistir do nível, sendo que ao morrer apenas retorna para o último *checkpoint*, pois ao reiniciar o nível ou o jogo poderia modificar drasticamente as reações do usuário, ocasionando frustração.

4.6 Análises complementares

Três baterias de teste foram feitas com grupos de amostra nos estudos de Rosa [13] e de Santos e Moraes [30]. No presente trabalho, os objetivos e as hipóteses desses estudos foram revisados com análises utilizando métodos estatísticos mais robustos, como apresentado e publicado nos anais do evento da 23^a Conferência Internacional sobre Interação Humano-Computador [103] e apresentado nas subseções seguintes.

Assim, análises complementares dos estudos anteriores foram executadas com codificação Python, utilizando os pacotes matplotlib, numpy, sklearn e pandas. Ademais, tiveram como cerne a robustez das métricas e os três tópicos principais:

1. Revisão das hipóteses propostas para comparação dos modelos de ADD de acordo com o estado de fluxo e a adequação da dificuldade ao jogador.
2. Identificação da correlação entre dados de desempenho e a dificuldade experienciada pelos jogadores.
3. Agrupamento de jogadores por dados de desempenho relacionados a perfis de jogadores.

Em cada bateria, o jogador poderia pausar o jogo, rever os controles e desistir de um nível. Além disso, as vidas eram infinitas e os níveis eram divididos em segmentos de mesma extensão delimitados por *checkpoints*, nos quais o progresso era salvo. A sessão de jogo era finalizada ao final de todos os níveis.

Na primeira bateria de Rosa [13], o jogo apresentava 12 níveis, analisando o uso de diferentes componentes do jogo (ou a junção destes) em cada nível. Participaram 20 universitários, com idades entre 18 e 24 anos e maior participação masculina (70%). Os jogadores tinham uma predileção por jogos com dificuldade fácil (45%) em comparação a normal (30%) ou difícil (25%).

Esperava-se determinar um balanceamento da dificuldade do jogo e analisar o método de estimativa da dificuldade proposto e implementado no jogo de plataforma *The Explorer: 2D*. Portanto, o objetivo foi verificar se a dificuldade estimada é capaz de se relacionar

com a dificuldade experienciada pelos jogadores. Em complemento, foi feita análise com modelos de regressão podem prever o valor da variável de dificuldade experienciada a partir dos dados de desempenho, inferindo a relação de uma variável dependente com variáveis independentes específicas.

Na segunda bateria de Rosa [13], o jogo apresentava 12 níveis com progressão de dificuldade em 3 conjuntos de 4 níveis. Para cada conjunto, seguiu-se o padrão: (1) nível sem a aplicação de ADD, (2) com ADD baseado no desempenho do jogador, (3) com ADD afetivo e (4) com ADD híbrido. Esse padrão foi repetido 3 vezes, considerando que os componentes e as dificuldades de base foram semelhantes em cada conjunto de níveis e que o ajuste poderia alterar a dificuldade geral de acordo com o modelo de ADD aplicado no nível.

Participaram 36 universitários, com idades entre 18 e 25 anos e maior participação masculina (61,1%). Os jogadores tinham uma predileção por jogos com dificuldade fácil (50%) em comparação a normal (33,3%) ou difícil (16,7%). Além disso, 16 dos 20 participantes da primeira bateria de Rosa participaram novamente, observando que novos níveis foram criados para testar a aplicação dos modelos ADD (por desempenho, afetivo e híbrido).

Pretendia-se comparar os 4 modelos do Sistema de Análise de ADD. Portanto, foram selecionados componentes semelhantes em cada conjunto e evitado caminhos exploratórios, a fim de não conter fatores particulares que pudessem influenciar diferentes experiências em um mesmo conjunto. A partir disso, foi definida a seguinte hipótese:

Hipótese 1 *Um sistema ADD híbrido com dados de desempenho e afetivo pode fornecer dificuldade adequada para o jogador.*

Por sua vez, foi feita uma análise de agrupamento a partir de dados brutos do jogo, fornecendo uma maneira de explorar a distribuição e a estrutura do conjunto de dados de desempenho e permitindo determinar se um método de reconhecimento de perfil de jogador pode ser usado para melhorar o mecanismo de ADD. Além disso, foi necessário selecionar os dados com maior valor de informação, de acordo com o objetivo de cada análise, e *outliers* de avaliações específicas foram identificados e removidos usando o z-score, o qual mostra quantos desvios padrão abaixo ou acima da média da população está uma pontuação bruta.

Além da medição da dificuldade, os jogadores podem ser categorizados em diferentes grupos. Essas categorias possibilitam a adaptação do conteúdo dos níveis com base nos perfis de jogador, aprimorando a lógica quando comparada às adaptações baseadas exclusivamente no desempenho. Nesse sentido, os dados foram analisados a partir de um modelo de reconhecimento do perfil de jogador, relacionada aos riscos assumidos pelo jogador e a sua habilidade. Ao considerar a recompensa do jogador em termos de satisfação

geral, uma divisão típica de estilos de gerenciamento seria busca de risco, risco neutro e avesso ao risco [11]. Esses três perfis de risco diferentes relacionam utilidade e recompensa. Os caçadores de risco ficam satisfeitos apenas com grandes recompensas e riscos mais elevados, enquanto os jogadores avessos ao risco preferem decisões de menor risco e recompensa.

Na bateria de testes de Santos e Moraes [30], o jogo também contou com 12 níveis, sendo executados de forma remota com os participantes baixando o jogo em suas máquinas e submetendo os resultados, sendo aleatorizado o modelo para cada voluntário e garantido que cada um jogasse apenas uma vez. As adaptações disponíveis foram: (1) jogo sem a aplicação de ADD, (2) com adaptação do tamanho de plataformas, (3) com adaptação da amplitude do salto e (4) com adaptação combinada das plataformas e da amplitude do salto.

155 voluntários participaram remotamente, com idades entre 15 e 65 anos, e com maior participação de homens (81,69%). 12 dos 155 voluntários foram descartados por não completarem a sessão de jogo e 11 por serem categorizados como *outliers*, desistindo dos níveis com menos de 10 segundos e com no máximo uma morte e, conseqüentemente, indicando que o jogador não se comprometeu com a tentativa. Cada jogador teve apenas uma adaptação de ADD aplicada (37 sem ADD, 29 com ADD de plataforma, 35 com ADD de salto e 31 com ADD combinado), totalizando 132 voluntários considerados para as análises. Além disso, os jogadores tinham uma predileção por jogos com dificuldade normal (51,41%) em comparação a difícil (32,39%) ou fácil (16,20%).

Nota-se que todos os níveis eram jogados integralmente por cada voluntário com uma única versão da adaptação do Sistema de Controle e que foi utilizado apenas o modelo baseado em desempenho. Todavia, haviam funções heurísticas distintas para cada adaptação testada. Nesse cenário, as seguintes hipóteses foram definidas:

Hipótese 2 *Um sistema ADD combinado com adaptação das plataformas e da amplitude do salto fornece uma dificuldade adequada para o jogador.*

Hipótese 3 *Um sistema ADD combinado com adaptação das plataformas e da amplitude do salto mantém o jogador em um estado de fluxo.*

Os testes foram validados pela aplicação de questionários durante a sessão de jogo. As questões aplicadas antes do início do jogo se referiam a dados sociodemográficos. Durante o jogo, verificava-se a dificuldade percebida em cada nível, sendo feita na primeira e na segunda baterias de Rosa. Enquanto isso, para a bateria de Santos e Moraes, não houve questionários entre níveis para não prejudicar o estado de fluxo. Ao final do jogo, analisava-se a dificuldade experienciada e se o jogador permaneceu em um estado de fluxo, para segunda bateria de Rosa e para bateria de Santos e Moraes, ou a determinação de

características que influenciam o desempenho do jogador e do desafio proporcionado por diferentes componentes, para primeira bateria de Rosa.

Em todos os casos de teste, foram coletados e salvos os dados de desempenho e, para primeira e segunda baterias de Rosa, os dados afetivos. As condições de teste garantiram que todos os participantes tivessem as mesmas informações do jogo e que os testes fossem executados em um mesmo ambiente sem contato externo para as baterias presenciais.

4.6.1 Balanceamento e verificação de medição de dificuldade

Na primeira bateria de testes de Rosa, a dificuldade de cada nível foi descrita numericamente pelos participantes por meio de um questionário após jogá-lo, conforme Figura 4.18. Os níveis testaram componentes distintos e suas interações, sendo possível identificar a proporcionalidade entre o valor normalizado calculado pelo algoritmo e o obtido pela média dos jogadores, entre 0 e 10. Por conseguinte, este método de estimação pode ser utilizado na aplicação do ADD. Porém, o nível 6 mostrou que o design do jogo fazia com que a dificuldade não variasse em conformidade. Nesse nível, o desafio era criado através do *design* proposto, enquanto o algoritmo contava apenas as posições e especificações dos componentes de plataforma para definição da dificuldade dos níveis.

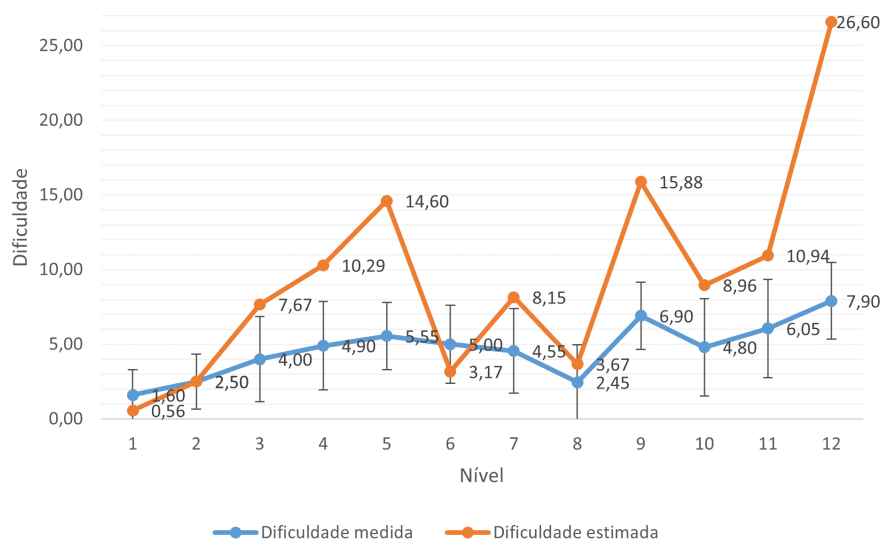


Figura 4.18: Comparação entre a média das dificuldades medidas pelo questionário e a dificuldade estimada pelo algoritmo desenvolvido em cada nível da primeira bateria de testes de Rosa (adaptado de [13]).

Houve pico nos níveis 5, 9 e 12 por terem *checkpoints* e dobrarem o tamanho do percurso a ser percorrido, trazendo assim maior dificuldade pelo acúmulo de desafios e componentes. Dessa forma, foi possível identificar que existe uma relação entre a quan-

tidade de componentes e a dificuldade percebida pelos jogadores. Além disso, observa-se que a modificação de componentes específicos altera a percepção de dificuldade do usuário, como visto com os níveis complementares 2 e 7 que alteravam a dificuldade apenas pelo aumento do tamanho das buracos.

Ao final de cada sessão de jogo, questionou-se sobre as ações que mais influenciavam o desempenho dos jogadores, sendo o maior objetivo de completar o nível (com 80% de frequência entre os participantes) em vez de completar os níveis rapidamente (25%). Além disso, houve uma baixa preocupação em explorar (30%) e derrotar o máximo de inimigos (30%). Assim, o ADD foi baseado em componentes mais gerais (como plataformas e buracos) e notou-se ser melhor testá-lo com caminhos lineares, limitando desvios para exploração.

Observou-se uma alta taxa de participantes concluindo os níveis, com decréscimo conforme a maior dificuldade dos níveis, e uma grande variação no tempo de permanência em cada nível, principalmente quando se analisa os níveis com componentes rígidos, conforme visto na Tabela 4.1. Ademais, desistências em um nível ocorriam quando haviam muitas mortes ou um alto índice de frustração do participante. Esses fatores são normalizados com a aplicação do ADD.

Nível	Completado (%)	Tempo Mínimo (s)	Tempo Máximo (s)	Tempo Médio (s)	Desvio Padrão
1	95	16,54	102,25	43,16	23,98
2	100	9,48	85,62	29,33	18,77
3	100	14,48	117,33	44,59	31,58
4	95	18,67	149,82	81,44	40,87
5	90	35,55	425,06	132,51	89,05
6	90	37,89	252,40	101,89	61,55
7	90	12,74	236,02	69,77	62,87
8	100	17,42	60,03	28,02	13,96
9	80	96,05	438,78	219,28	99,27
10	80	14,61	226,31	74,42	66,58
11	85	39,23	320,63	110,20	62,36
12	55	45,30	603,66	285,75	145,85

Tabela 4.1: Porcentagem de participantes que concluíram, tempo mínimo, tempo máximo, tempo médio e desvio padrão relacionados à permanência em cada nível da primeira bateria de testes de Rosa (adaptado de [13]).

Correlação entre dados de desempenho e dificuldade

Para a análise de regressão, a variável resposta (Y) foi definida como a dificuldade vivenciada por cada participante, informada ao final de cada um dos 12 níveis da primeira

bateria de testes de Rosa. Ademais, os dados de desempenho foram vistos como variáveis preditoras (X), considerando a quantidade de mortes, o tempo jogado e a quantidade de saltos para as análises. A Tabela 4.2 mostra o coeficiente de determinação (R^2) encontrado para as diferentes variáveis com e sem transformação logarítmica. Uma segunda análise foi feita comparando apenas os níveis que foram concluídos pelo jogador.

X	R^2	R^2 (log)	R^2 (log completado)
Tempo jogado	0,3339	0,5151	0,5032
Tempo andando para o final	0,3339	0,4171	0,4250
Saltos	0,2645	0,4324	0,4399
Saltos de sucesso	0,2153	0,3659	0,3717
Saltos falhos	0,1998	0,3428	0,3313
Taxa de morte	0,1310	0,1296	0,1480

Tabela 4.2: Coeficiente de determinação (R^2) encontrado para cada uma das variáveis preditoras declaradas (X), considerando a dificuldade como a variável de resposta (Y). Sem transformação logarítmica, com transformação logarítmica e com transformação logarítmica desconsiderando níveis que não foram completados pelo jogador na primeira bateria de testes de Rosa.

A dificuldade, informada por meio de questionário para cada nível, variava entre 0 e 10. No entanto, todas as variáveis preditoras têm uma ampla gama de dados, porque dependem do desempenho de cada jogador. Portanto, a transformação logarítmica foi aplicada para normalizar os dados.

O maior coeficiente de determinação correspondeu ao tempo jogado como variável preditora. Pode-se prever a dificuldade experienciada pelos jogadores com 51,51% da variação explicada pela dependência da variável do tempo jogado ao usar o modelo de regressão baseado em mínimos quadrados ordinários. Além disso, o coeficiente de determinação corrigido, de acordo com o número de características de entrada, foi de 0,513. A resposta prevista é mostrada na Equação 4.32, sendo Y a dificuldade e X o tempo jogado, e o modelo de regressão na Figura 4.19. Como o valor-p (abaixo de 0,00001) é menor do que o nível de significância de 0,05, a hipótese nula pode ser rejeitada e os dados fornecem evidências suficientes para concluir que o modelo de regressão se ajusta aos dados melhor do que o modelo sem variáveis independentes.

$$Y = -5,5142 + 5,5731 * X \quad (4.32)$$

A regressão linear múltipla também foi testada para a avaliação adicional de possíveis modelos de regressão a partir dos diferentes dados de desempenho. Houve melhora no coeficiente de determinação registrado, especialmente ao utilizar a transformação logarítmica aplicada aos dados. Portanto, o melhor coeficiente correspondeu à regressão

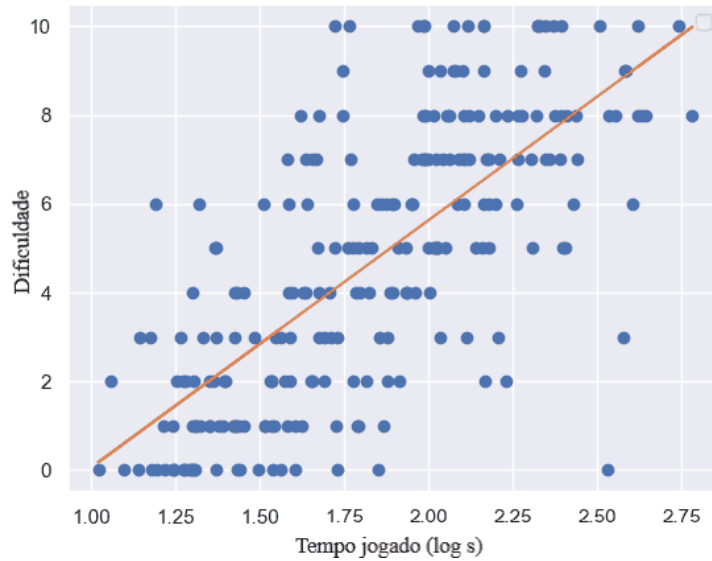


Figura 4.19: Modelo de regressão para o tempo jogado como variável preditora (X) e dificuldade experienciada como variável de resposta (Y) a partir de dados da primeira bateria de testes de Rosa.

linear múltipla com as variáveis preditoras de tempo jogado e quantidade de mortes, com valor-p abaixo de 0,00001 e a resposta prevista apresentada na Equação 4.33, sendo Y a dificuldade, X_1 o tempo jogado e X_2 a quantidade de mortes.

$$Y = -5,2450 + 5,3166 * X_1 + 0,7523 * X_2 \quad (4.33)$$

Mesmo com regressão múltipla, os resultados ainda têm um baixo coeficiente de determinação (de 0,5206) ao prever a dificuldade por meio de dados de desempenho. Logo, informações úteis podem ser obtidas a partir do modelo de regressão, como o fato de um tempo mais longo em um nível estar relacionado a uma percepção de dificuldade mais alta do jogador. Todavia, esse modelo de regressão não pode ser utilizado para predição de dificuldade, sendo possível aplicar outros métodos de aprendizado supervisionado, como redes neurais.

4.6.2 Comparação entre modelos por desempenho, afetivo e híbrido

Na segunda bateria de testes de Rosa, um teste de análise de variância foi aplicado para verificar as diferenças na distribuição dos modelos de ADD. Com isso, os dados de desempenho foram testados com o teste não paramétrico de Friedman, que não pressupõe a independência das observações, pois todos os voluntários jogaram para todos os modelos de ADD aplicados durante o jogo e os dados não eram normalmente distribuídos. As

análises foram feitas para cada caso, comparando os quatro modelos para conjuntos de nível fácil (1-4), normal (5-8) e difícil (9-12). Os dados foram verificados com o teste de Shapiro-Wilk, tendo valores-p significativos e não aceitando a hipótese nula de que a população era normalmente distribuída. Ademais, foi observado com o teste de Bartlett que as amostras das populações tinham variâncias iguais para a maioria dos dados de questionários e não tinham para os dados de desempenho.

Como visto na Figura 4.20, os jogadores descreveram numericamente a dificuldade dos níveis para três conjuntos de quatro níveis. A partir disso, foi possível observar que o valor médio cresce de acordo com a progressão dos níveis fácil e médio, o que é esperado tendo em vista a dificuldade de estar aumentando nos jogos de plataforma e no algoritmo de ajuste dos níveis para seguir esse padrão. Porém, há pico nos níveis 5 e 9 em relação ao padrão de crescimento anterior, pois não há aplicação do ADD nos mesmos.

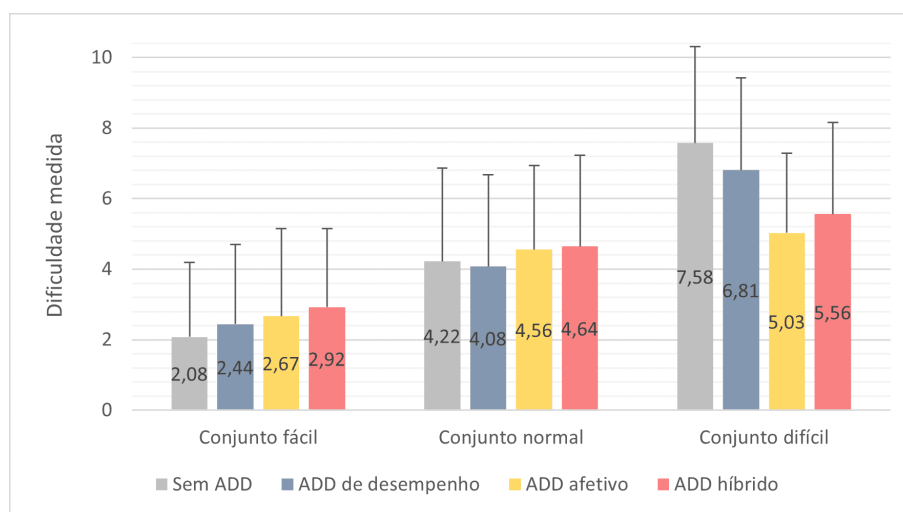


Figura 4.20: Valor médio e desvio padrão das dificuldades medidas pelo questionário em cada nível da segunda bateria de testes de Rosa, dividida nos três conjuntos de dificuldade testados e agrupados pelos quatro modelos (adaptado de [13]).

Conforme a progressão do jogo, mais dados eram capturados e ocorria uma melhor estimativa do desempenho e do estado afetivo do jogador. Assim, nos conjuntos de nível normal e difícil, um maior desvio padrão é observado quando a aplicação do ADD não ocorre. Além disso, a dificuldade desses níveis se destaca em relação aos imediatamente seguintes (6 e 10), pois estes se adequam ao desempenho do jogador em níveis anteriores, incluindo o nível sem ADD, e assim diminuem a dificuldade para a média dos jogadores.

Os níveis com menor percentual para serem completados em cada conjunto foram aqueles sem a aplicação do ADD (níveis 5 e 9 com 94,4% e 63,9%, respectivamente). Além disso, ocorreram picos nos níveis afetivo e híbrido, enfatizando que todos os participantes completaram os níveis com ADD afetivo. O menor percentual de conclusão para níveis

com ADD foi no nível 10 baseado em desempenho (80,5%), o qual ainda é superior ao menor percentual sem ADD no nível 9 (63,9%). A tabela 4.3 exibe os dados para a frequência de níveis concluídos e os tempos despendidos em cada nível.

Nível	Completado (%)	Tempo Mínimo (s)	Tempo Máximo (s)	Tempo Médio (s)	Desvio Padrão
1	100	20,59	177,18	58,35	32,37
2	100	18,39	66,93	38,39	12,74
3	100	22,62	92,68	39,28	13,28
4	100	20,45	75,18	39,41	13,34
5	94,4	19,35	473,24	101,52	90,82
6	97,2	15,19	133,91	53,64	26,17
7	100	35,47	178,37	85,14	36,80
8	100	23,75	210,43	61,02	36,08
9	63,9	45,02	698,00	234,64	139,71
10	80,5	47,32	229,03	121,75	46,76
11	100	27,68	157,80	61,14	28,68
12	94,4	57,33	161,38	99,54	25,84

Tabela 4.3: Porcentagem de participantes que concluíram, tempo mínimo, tempo máximo, tempo médio e desvio padrão relacionados à permanência em cada nível da segunda bateria de testes de Rosa (adaptado de [13]).

Além da taxa crescente de jogadores completando os níveis, considerando os dados de desempenho, uma gama mais ampla de diferença de medianas dos modelos é estatisticamente significativa, apenas não para taxa de mortalidade e contagem de saltos em níveis fáceis, considerando um nível de significância de 0,05. A Tabela 4.4 apresenta um recorte dos casos para o valor-p dos dados de desempenho medidos com o teste de Friedman.

Dificuldade	Dados	Valor-p
Fácil	Tempo jogado	0,0005
Médio	Taxa de morte	1,438e-10
Médio	Saltos falhos	2,5361e-06
Médio	Tempo jogado	1,3800e-05
Difícil	Taxa de morte	5,8626e-10
Difícil	Saltos	2,0277e-10
Difícil	Saltos falhos	2,0782e-13
Difícil	Tempo jogado	1,5065e-13

Tabela 4.4: Valor-p medido a partir do teste de Friedman, de acordo com os dados de desempenho para cada conjunto de níveis por dificuldade para segunda bateria de testes de Rosa.

Ademais, os dados de desempenho significativamente diferentes mais consistentes eram o tempo de jogo do participante em cada nível, mostrado na Figura 4.21. Nesse cenário,

observou-se a dispersões entre distintos modelos dentre de um mesmo conjunto de níveis. Assim, as estatísticas dos jogadores mostram um melhor desempenho com os modelos de ADD, principalmente considerando um menor tempo de finalização do jogo e com menor dispersão.

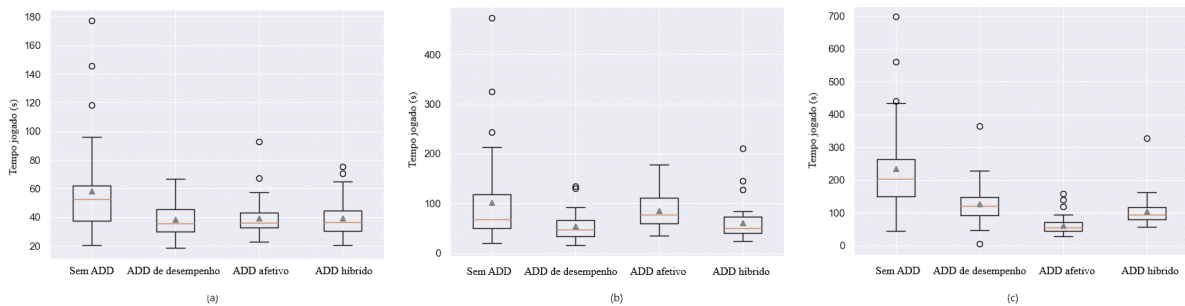


Figura 4.21: Diagrama de caixa para o tempo jogado, respectivamente, para os conjuntos de níveis (a) fácil, (b) normal e (c) difícil, comparando os quatro modelos na segunda bateria de testes de Rosa.

O desvio padrão em jogos sem ADD foi consideravelmente maior, pois não se adequam a cada jogador e conseqüentemente trazem uma variação maior no desempenho de jogadores distintos em tais níveis. Na maioria dos casos, os tempos médios em níveis com diferentes modelos de ADD são menores e com menor dispersão, havendo um pico negativo no desvio padrão no ADD de desempenho nos conjuntos de dificuldade fácil e normal e no ADD híbrido no conjunto difícil.

Além disso, ao final de cada nível, o jogador era questionado sobre o nível que apresentava uma dificuldade mais adequada entre o atual e o anterior, demonstrado na Figura 4.22. Como cada nível possuía um modelo diferente, os modelos puderam ser comparados e tiveram a seguinte ordem de melhor percepção pelos jogadores: ADD híbrido, ADD afetivo, ADD de desempenho e sem ADD. Observa-se que, é mais fácil notar uma diferença entre os modelos para os níveis difíceis, devido às maiores disparidades nas habilidades do jogador do que em níveis fáceis e normais. Todavia, a percepção do jogador em diferenciar a dificuldade de adequação diminui nas séries de nível superior, devido ao aumento da dificuldade.

Considerando os dados dos questionários e de desempenho apresentados acima, a análise correspondente pode validar a Hipótese 1. Isto é especialmente observado na redução da dispersão, melhores resultados de desempenho dos jogadores e na percepção demonstrada do jogador sobre a dificuldade experienciada com cada modelo. Portanto, um sistema ADD híbrido tem uma adaptação de dificuldade adequada para o jogador. Ademais, observa-se que testes de comparação devem ser feitos separadamente, testando apenas um modelo por participante, para medir a diferença entre os modelos por meio de questionários e manter o jogador em estado de fluxo.

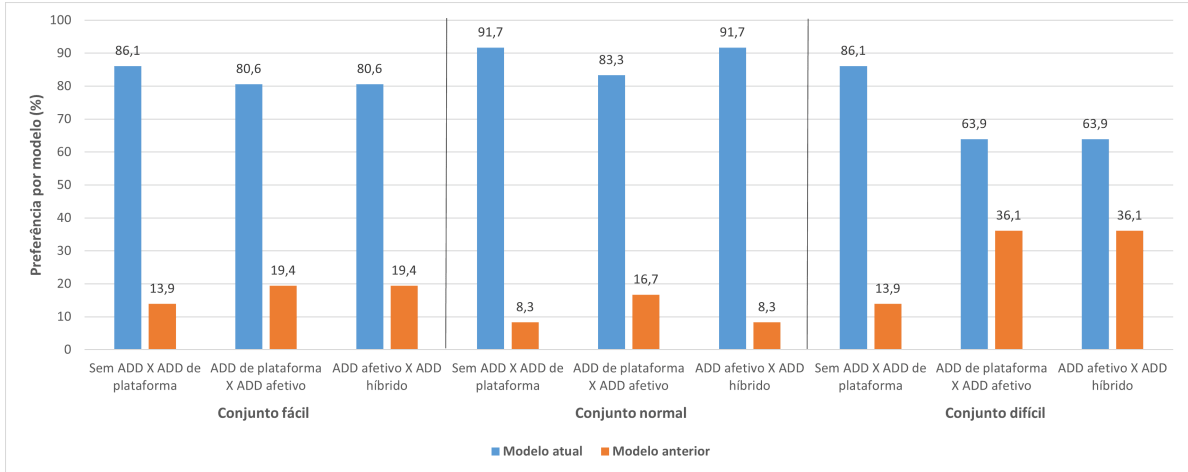


Figura 4.22: Frequência de respostas sobre a dificuldade ser mais adequada no nível atual ou anterior com frequências agrupadas por conjuntos de dificuldade, questionadas ao final dos níveis da segunda bateria de testes de Rosa (adaptado de [13]).

Reconhecimento do perfil de jogador

A duas baterias de Rosa foram analisadas para identificar os dados com maior correlação ao valor de dificuldade experienciada e ao perfil de jogador. Utilizou-se o algoritmo de agrupamento *K-means* com $K = 3$ (*clusters* de risco e habilidade). Observa-se que a transformação logarítmica foi aplicada para normalização e consideramos apenas dados relacionados ao métodos de ADD, como tempo, saltos e mortes por queda, descartando mortes por inimigos e armadilhas, as quais não são consideradas no método. Desse modo, a Figura 4.23 está relacionada ao agrupamento por habilidade e perfil de risco. Os melhores cenários encontrados para a análise de agrupamento pretendida envolviam quantidade de saltos, quantidade de saltos falhos, tempo jogado e tempo de resposta (tempo jogado dividido pela quantidade de saltos).

Outros métodos de agrupamento foram testados, mas não houve resultados positivos como o *K-means*. O algoritmo *Mean-Shift* ajuda a encontrar as áreas densas de pontos de dados e o algoritmo Hierárquico Aglomerativo não é sensível à escolha da métrica de distância. Todavia, ambos possuem um número dinâmico de *clusters* e nossa análise pretende comparar um número fixo de *clusters* com os perfis esperados. Além disso, o teste de Hopkins para tendência de *cluster* foi aplicado a cada método (maior quando próximo de 1), encontrando melhores resultados para *K-means* neste conjunto de dados, como indicado na Tabela 4.5.

Observa-se que a quantidade de saltos pode se relacionar ao perfil de risco (mais saltos dados por jogadores que se arriscam), a quantidade de saltos falhos à habilidade e o tempo de resposta (tempo jogado dividido pela quantidade de saltos) à habilidade de resolver aos desafios rapidamente e o perfil de risco (baixo tempo de resposta para aqueles que

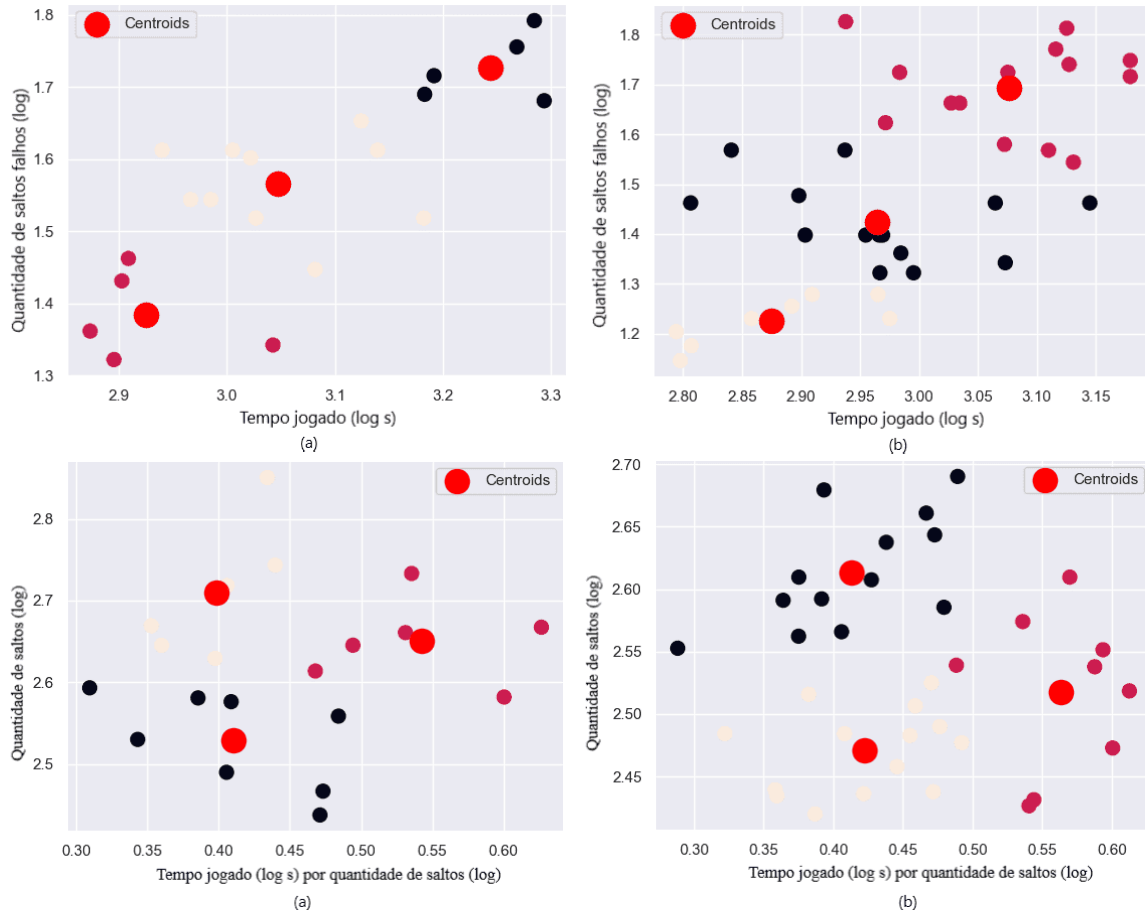


Figura 4.23: Resultados do agrupamento *K-means* com 3 *clusters* para (a) primeira e (b) segunda baterias de testes de Rosa com quantidade de saltos falhos e tempo jogado ou quantidade de saltos e tempo jogado dividido pela quantidade de saltos como parâmetros, respectivamente.

Dados	K-means	Mean-Shift	Aglomerativo
Tempo jogado (s) dividido pela quantidade de saltos X Quantidade de saltos	0,8722	0,6572	0,6333
Tempo jogado (s) dividido pela quantidade de saltos X Quantidade de saltos falhos	0,8722	0,6777	0,6332

Tabela 4.5: Valor de Hopkins encontrado para distintos métodos (*K-means*, *Mean-Shift* e Aglomerativo), de acordo com um subconjunto de dados de desempenho capturados na segunda bateria de testes de Rosa.

assuem riscos). Desse modo, os *clusters* gerados podem agrupar jogadores de acordo com o perfil de risco e podem fornecer dados úteis para o algoritmo de ADD. Todavia, análises mais robustas são necessárias para seleção dos atributos, incluindo outros dados de desempenho e métodos que correlacionem diretamente com perfis específicos.

4.6.3 Comparação entre adaptação de plataforma, salto e combinado

Na bateria de testes de Santos e Moraes, foram feitas perguntas ao final das sessões de jogo para verificar se o jogador estava em um estado de fluxo e se a dificuldade era adequada, respectivamente apresentado nas Figuras 4.24 e 4.25. Nesse sentido, os voluntários relataram as adaptações combinadas e de plataforma como as melhores em média para manter o estado de fluxo do jogador, considerando o foco no jogo, a perda de tempo e as ações automáticas. Os casos em que nenhum ADD foi aplicado vieram a seguir, com a adaptação apenas no salto tendo uma avaliação inferior.

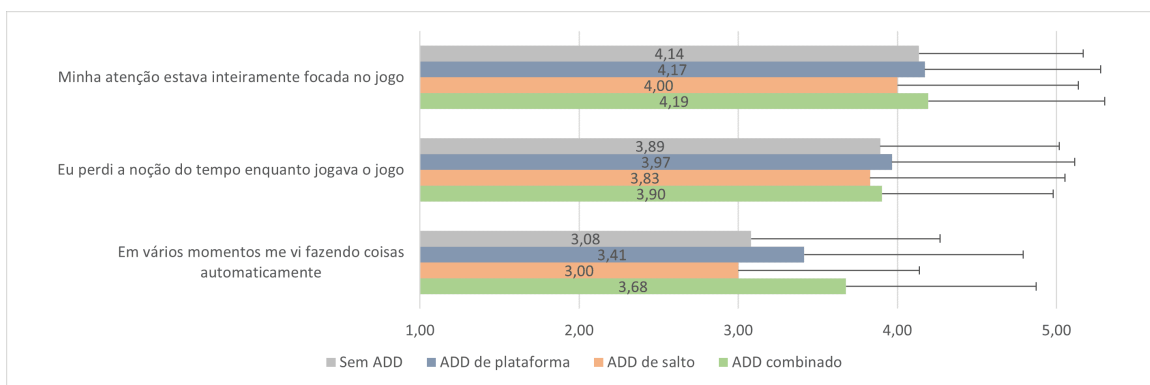


Figura 4.24: Valor médio e desvio padrão das questões relacionadas ao estado de fluxo na bateria de testes de Santos e Moraes, agrupadas pelos quatro tipos de adaptação.

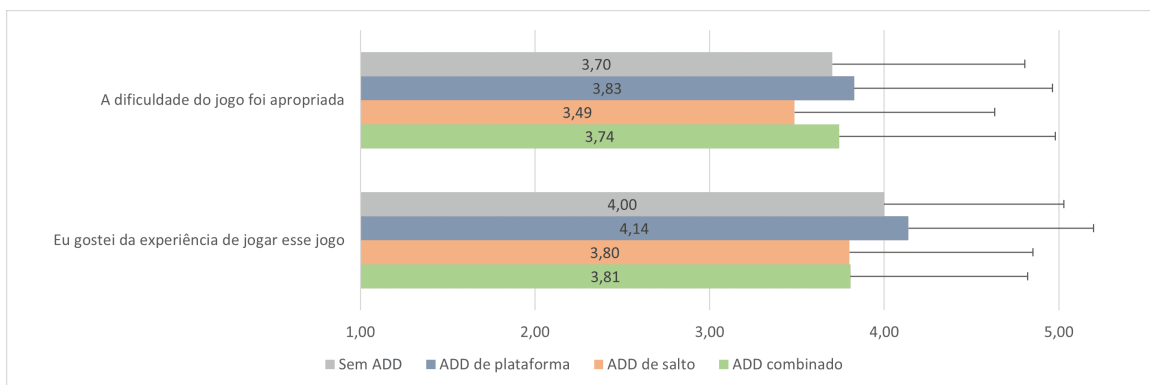


Figura 4.25: Valor médio e desvio padrão das questões relacionadas à adequação ou dificuldade na bateria de testes de Santos e Moraes, agrupadas pelos quatro tipos de adaptação.

A dificuldade encontrada foi a mais adequada para a adaptação da plataforma, seguida da adaptação combinada e com menor resultado para o jogo sem ADD. Além disso, pela qualidade da experiência percebida pelos usuários durante o jogo, o ADD aplicado apenas às plataformas teve um melhor desempenho em relação aos outros modelos. Percebe-se

também que jogar sem ADD ainda rendia uma experiência melhor do que o ADD aplicado ao salto ou combinado.

Os dados foram testados com o teste não paramétrico de Kruskal-Wallis, pois não apresentavam distribuição normal e foi garantido que cada voluntário jogasse apenas com uma adaptação do ADD, garantindo a independência das observações. Para um nível de significância de 0,05, as diferenças entre as medianas não foram estatisticamente significativas para a maioria dos dados do questionário.

Como os níveis do jogo são projetados para crescer com o desempenho do jogador, supõe-se que níveis com extremos mais de dificuldade podem ser necessários para analisar a diferença quando os modelos aplicados em relação às percepções do jogador. Nesse sentido, pode existir uma covariância entre a construção dos níveis e o modelo de ADD, com os níveis variando de forma fixa e o ADD variando em cima disso. Para evitar essa situação, testes futuros precisam incluir níveis revisados para fornecer maior variabilidade de adaptações.

Assim como na segunda bateria de Rosa, a divergência de dados mais consistente para os diferentes níveis foi o tempo de jogo para completar cada nível, apresentado na Tabela 4.6. O tempo de jogo em todos os níveis teve menor média e menor desvio padrão para o jogo com adaptações, com menor média para a adaptação do salto e menor desvio padrão para a adaptação da plataforma. Portanto, as adaptações foram capazes de diminuir a variação no desempenho de jogadores distintos.

Adaptação	Tempo jogado		
	Média	Mediana	Desvio Padrão
Sem ADD	1164,00	808,39	1110,87
ADD de plataforma	1038,46	884,63	517,84
ADD de salto	1012,95	830,73	689,65
ADD combinado	1080,32	924,24	660,62

Tabela 4.6: Média, mediana e desvio padrão do tempo para completar o jogo em cada adaptação na bateria de testes de Santos e Moraes.

As adaptações do ADD visam diminuir a dispersão dos dados de desempenho, como o tempo jogado e a taxa de mortalidade. Os resultados mostraram melhor adequação quanto à preferência do jogador por jogos fáceis. Assim, as Tabelas 4.7 e 4.8 apresentam os valores referentes ao tempo jogado e a quantidade de mortes por queda (em abismos ou ácidos) para jogadores com predileção pela dificuldade fácil.

Ao considerar os dados de desempenho para jogadores com preferência por jogos fáceis, as diferenças desses parâmetros foram estatisticamente significativas ao usar o teste de Kruskal-Wallis com 0,5 de nível de significância, comparando o jogo sem ADD com cada adaptação distinta e obtendo valores-p entre 0,018 e 0,023. Nota-se que o jogo tem um

Adaptação	Tempo jogado		
	Média	Mediana	Desvio padrão
Sem ADD	3359,44	2776,09	1900,83
ADD de plataforma	1471,35	1166,59	842,73
ADD de salto	1119,11	1212,23	236,83
ADD combinado	1230,49	1257,71	267,89

Tabela 4.7: Média, mediana e desvio padrão do tempo para completar o jogo em cada adaptação com jogadores que preferem a dificuldade fácil na bateria de testes de Santos e Moraes.

Adaptação	Quantidade de mortes		
	Média	Mediana	Desvio padrão
Sem ADD	50	85	25,02
ADD de plataforma	30	68	21,98
ADD de salto	31	75	14,83
ADD combinado	33	73,5	10,15

Tabela 4.8: Média, mediana e desvio padrão da quantidade de mortes por queda em cada adaptação com jogadores que preferem a dificuldade fácil na bateria de testes de Santos e Moraes.

limite de dificuldade proporcionada ao aumentar a distância entre plataformas, o que pode justificar o impacto maior em jogadores que preferem um jogo fácil. No caso, jogadores com preferência por jogos difíceis podem precisar de desafios complexos além da dinâmica básica do salto entre plataformas.

As adaptações da plataforma e do composto geraram uma melhor experiência em relação ao estado de fluxo, mas sem diferença significativa. Portanto, não foi possível confirmar a Hipótese 3 ao verificar se o ADD combinado foi capaz de melhorar a manutenção do jogador em um estado de fluxo ao usar o jogo sem ADD como base. Por outro lado, o sistema combinado de ADD foi capaz de adequar a dificuldade aos jogadores, diminuindo a dispersão de desempenhos e confirmando a Hipótese 2. O resultado foi positivo principalmente para os jogadores que preferem a dificuldade fácil, mostrando que o sistema pode ser melhorado adaptando o jogo de acordo com características dos perfis de jogadores.

Capítulo 5

Proposta do Ajuste Dinâmico de Dificuldade

Este capítulo apresenta a proposta do trabalho, os desafios enfrentados e as soluções aplicadas.

5.1 Proposta

O presente trabalho trata do desenvolvimento de um sistema de Ajuste Dinâmico de Dificuldade (ADD) para jogos de plataforma. Para isso, quatro etapas foram definidas:

1. Análise de dados de testes anteriores;
2. Aprimoramento de métodos de estimação de dificuldade, aplicação de adaptações em jogo e medição do estado de fluxo;
3. Definição de técnica para reconhecimento de perfis de jogadores; e
4. Aplicação de Ajuste Dinâmico de Dificuldade por meio de perfis de jogadores categorizados.

Em primeira instância e considerando métricas mais robustas de análise de dados, pretende-se revisar e complementar os resultados dos experimentos feitos nos estudos de Rosa [13] e Santos e Moraes [30], apresentada na Seção 4.6.

Após isso, objetiva-se investigar a eficiência dos métodos computacionais para reconhecer os perfis de jogador e a dificuldade experienciada, correlacionando estatisticamente às variáveis de desempenho extraída das sessões de jogo conforme as ações dos jogadores. Em complemento, pretende-se aprimorar a estimativa de dificuldade e as métricas definidas para aplicação de ajustes. Nesse sentido, os níveis foram planejados para conter

componentes genéricos de jogos de plataforma de forma a averiguar suas dificuldades. Logo, também é necessário o teste da predição do perfil do jogador pelos dados de jogo em comparação aos questionários, definindo parâmetros que possam ser utilizados pelo ADD.

Em seguida, almeja-se aplicar métricas mais robustas ao ADD baseado em desempenho, verificando se o sistema mantém o jogador em um estado de fluxo e possui um melhor resultado na adequação de dificuldade ao usuário, incluindo uma melhor definição do contexto para medição do estado de fluxo. Para isso, os testes de comparação de modelos do Sistema de Controle do ADD precisam de contextos que estimulem diferentes reações, testando as métricas e os instrumentos de percepção.

No caso, experimentos com propostas similares aos testes de Santos e Moraes [30] são importantes que sejam realizados. Mesmo que houvesse dados significativos em relação a uma menor dispersão do desempenho dos jogadores, as respostas de questionários relativas ao estado de fluxo não tiveram diferenças significativas entre os distintos modelos de Sistema de Controle de ADD testados. Nesse sentido, é necessária a atualização dos questionários, especialmente referente a escala de avaliação do jogador e seu estado de fluxo. Nota-se que os participantes poderiam estar condicionados a responder em determinadas faixas da escala Likert e que o estímulo dos níveis do jogo já buscavam a progressão e imersão do jogador, sem causar variância significativa.

Protocolos para aplicação do Sistema de Controle precisam ser definidos e os testes podem incluir níveis com maior variabilidade do uso de componentes e extremos de dificuldade. Com isso, espera-se aumentar a discrepância de percepções da experiência de dificuldade e diversão dos jogadores ao forçar reações do usuário. A elaboração dos níveis com extremos de dificuldade evita a co-variância entre a dificuldade crescente dos níveis e a adaptação dos modelos de ADD que pode ter ocorrido nos testes anteriores e gerado baixa variabilidade resultados dados de questionário.

Ao final, objetiva-se utilizar Aprendizado de Máquina para reconhecimento do perfil de jogador e estimação da dificuldade, aprimorando o Sistema de Análise do ADD. Assim, averigua-se a capacidade um sistema de ADD integrado com Aprendizado de Máquina e utilizando características do perfil do jogador em manter o jogador em um estado de fluxo e diminuir a dispersão entre o desempenho de distintos jogadores.

Estudos associam dados de desempenho com Aprendizado de Máquina [29], porém poucos relacionam com o perfil do jogador [11]. Observa-se que, ao executar o ajuste associando a dificuldade experienciada pelo jogador ao seu perfil, é possível que as adaptações sejam feitas de forma mais adequada. Dependendo do perfil, a dificuldade desejada pelo jogador pode diferenciar em méritos como resiliência e aversão a riscos.

5.2 Aprimoramento do método

Em vista do objetivo de robustez do modelo de ADD, o diagrama cíclico apresentado na Seção 4.4 foi revisto, como visto na Figura 5.1. Nesse contexto, assim como o Sistema de Monitoramento pode ser dividido em medidas de desempenho (a partir de dados brutos do jogo) e afetivas (a partir de dados fisiológicos), o Sistema de Controle se divide em adaptações de mundo (como plataformas e buracos) e adaptações de características do avatar (como amplitude do salto). As adaptações de mundo também podem conter mudanças de qualquer outro componente do jogo, como inimigos e objetos de auxílio, enquanto as adaptações de características do avatar podem conter mudanças das demais configurações do avatar, como velocidade horizontal e quantidade de vidas.

Além disso, para novos experimentos, foi definido que o Sistema de Análise pode ser dividido em diferentes métodos, como o uso de heurísticas ou de Aprendizado de Máquina para determinar o que será adaptado no jogo. Observa-se que o Sistema de Análise permite que as adaptações sejam feitas não só considerando a habilidade do jogador, mas utilizando características específicas do jogador para uma melhor adequação da dificuldade.

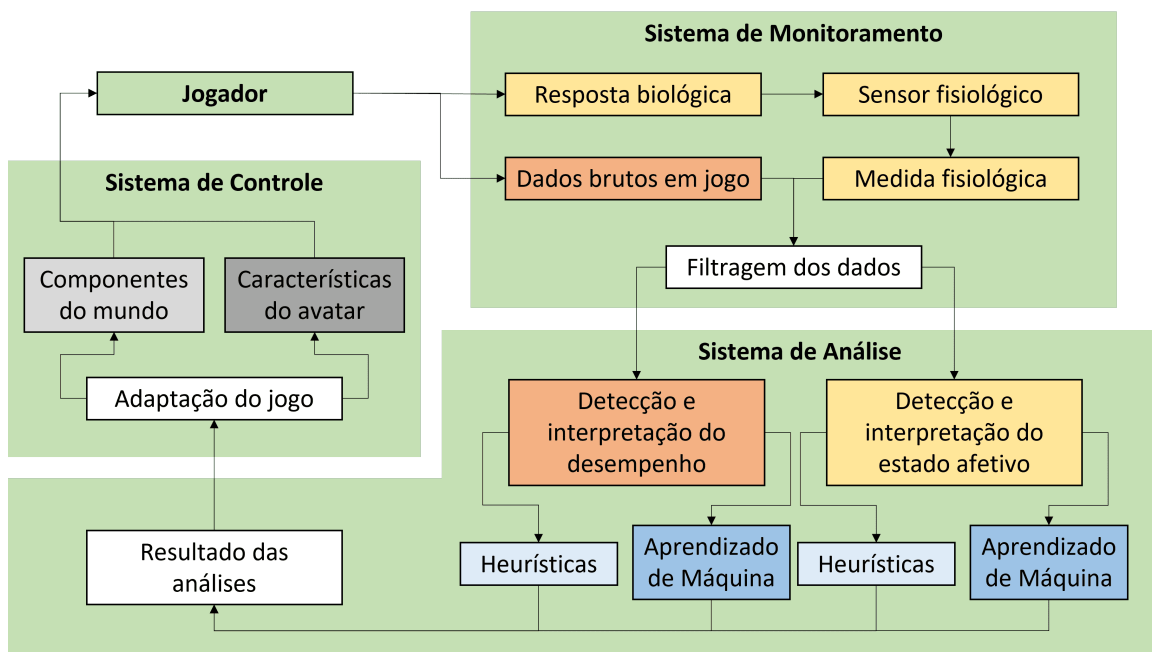


Figura 5.1: Diagrama cíclico completo do Ajuste Dinâmico de Dificuldade orientado pelo sistema. Os sistemas estão definidos em verde, as ações referentes ao desempenho do jogador em laranja e a afetividade em amarelo, as diferentes opções de análise em tons de azul e as diferentes opções de adaptação em tons de cinza.

De modo complementar, os questionários, detalhados no Apêndice A, foram atualizados após revisão da literatura. Foram utilizadas as bases de dados *IEEE Xplore*

*Digital Library*¹, *Association for Computing Machinery (ACM)*², *Google Scholar*³ e *Research Gate*⁴. Mais de 150 artigos relacionados foram analisados, sendo que mais de 80 estudavam diretamente a aplicação do ADD.

Com isso, foram incluídas perguntas no questionário demográfico para determinação do perfil do jogador pelo modelo Hexad, determinado por [104], [105] e [106], e modificadas as perguntas do questionário final referente ao estado de fluxo, determinado por [107], [108] e [109].

Os estudos que serviram como base para o questionário executaram análises quantitativas para validação da correlação entre as perguntas e os fatores analisados. Ademais, a faixa de valores mudou de 1 a 5 para 1 a 7 para perguntas na escala Likert, considerando não ter variabilidade significativa em testes anteriores para comparação do estado de fluxo entre distintos modelos.

5.2.1 Modificações do Sistema de Controle

Considerando o aprimoramento da robustez do método de ADD, o Sistema de Controle foi revisto para definir métricas de adaptação que mantenham padrões de *design*, evitem conflito na construção do mapa e impeçam a existência de componentes de plataforma impossíveis de serem alcançados. Nesse sentido, algumas regras de adaptação de buracos e plataformas, ilustradas na Figura 5.2, foram especificadas e programadas para o jogo:

- Plataformas e buracos são adaptados aumentando ou diminuindo a distância entre a origem e o destino nos eixos x e y, considerando as origens e os destinos descritos na Subseção 4.4.1.
- Adaptações só ocorrem quando a dificuldade resultante segue o padrão desejado de modificação, aumentando ou diminuindo, e que o componente se mantenha alcançável em um salto.
- Os dados de ponto inicial, final e tamanho da plataforma devem ficar salvos para permitir definir os limites da adaptação. Para plataformas móveis, ilustrado na Figura 5.3, também deve estar salvo o caminho e velocidade da movimentação. Caso a origem ou o destino seja uma plataforma móvel, o ponto inicial, final e tamanho são atualizados em conformidade ao movimento.
- O limite inferior para as adaptações do componente de destino é definido de acordo com o eixo e a direção do salto:

¹<https://ieeexplore.ieee.org/>

²<https://dl.acm.org/>

³<https://scholar.google.com.br/>

⁴<https://www.researchgate.net/>

- No eixo vertical (y):
 - * Para destinos iguais ou superiores a origem (Superior) e para destinos iguais ou inferiores a origem (Inferior), o limite inferior é a posição mais baixa do componente de destino, para que o componente sempre continue existindo e não afete a caminho para outros componentes.
- No eixo horizontal (x):
 - * Para destinos iguais ou após a origem (Direita) e para destinos iguais ou antes da origem (Esquerda), o limite inferior é a última posição do componente de origem, na qual o avatar poderá entrar em contato antes do salto.
- O limite superior para as adaptações do componente de destino é definido de acordo com o eixo e a direção do salto:
 - No eixo vertical (y):
 - * Para destinos iguais ou superiores a origem (Superior), o limite superior é a posição mais alta entre a origem do salto e altura máxima do salto.
 - * Para destinos iguais ou inferiores a origem (Inferior), o limite superior é a posição mais alta do componente de origem, na qual o avatar poderá entrar em contato antes do salto.
 - No eixo horizontal (x):
 - * Para destinos iguais ou após a origem (Direita), o limite superior é a posição mais distante entre a origem do salto, a distância horizontal máxima do salto e a última posição do componente de destino, para que o componente sempre continue existindo e não afete a caminho para outros componentes.
 - * Para destinos iguais ou antes da origem (Esquerda), o limite superior é a posição mais distante entre a origem do salto, a distância horizontal máxima do salto e a primeira posição do componente de destino, para que o componente sempre continue existindo e não afete a caminho para outros componentes.
 - Para plataformas atravessáveis, os limites podem ser estendidos horizontalmente pela extensão da origem, em vista do avatar conseguir atravessar a plataforma de destino ainda que esteja paralela verticalmente a origem.
- A adaptação de dificuldade de plataformas móveis deve considerar somente a dificuldade espacial, caso somente esta esteja sendo adaptada, conforme descrito na Seção 4.2. Caso a dificuldade temporal também esteja sendo adaptada, ambas devem ser

consideradas. Isso deve ser feito para permitir uma maior amplitude de adaptações nesse tipo de componente.

- Plataformas podem ser bloqueadas para serem adaptadas horizontal ou verticalmente para evitar situações específicas. Por exemplo, no contexto do jogo *The Explorer: 2D*, plataformas atravessáveis só possuem um ladrilho de altura, não devendo ser adaptadas no eixo vertical para que o avatar não fique preso durante um salto.
- Ao executar uma adaptação, o ponto de origem pode ser redefinido analisando as dimensões atualizadas do componente de origem.
 - A origem é um ponto que pode ser variado de acordo com as adaptações de outras plataformas e buracos. Logo, considerando caminhos lineares, as adaptações devem ser feitas em ordem de quando os componentes aparecem. Em cenários não lineares, as adaptações devem ser feitas conforme a ordem de caminhos possíveis para o jogador. Ou seja, seguindo o conceito de um grafo, componentes ficam mapeados com a informação de quais outros componentes os precedem nos caminhos previstos para o jogador e as adaptações devem seguir essa ordem do ponto inicial ao final do nível.
- A adaptação de plataformas deve ser restringida embaixo do ponto de início, de *checkpoints* e de portais de saída do nível ou em posições que já possuem um componente.
- As adaptações priorizam igualmente mudanças nos eixos x e y, independente da ordem de execução, ao considerar a dificuldade de componente como única e variável conforme a probabilidade de sucesso do salto, conforme descrito na Seção 4.2.
- As adaptações devem adicionar ladrilhos em toda a sua extensão para que o componente fique visualmente coerente. Por exemplo, ao adicionar ou remover ladrilhos no eixo x para um componente, a ação deve ser feita em todos os pontos da extensão original do eixo y.

5.2.2 Atualização do método de medição de dificuldade de plataformas

Para o aprimoramento do Sistema de Controle, também foi necessário atualizar e complementar as fórmulas para estimação da dificuldade de componentes de plataforma, descrita na Seção 4.2. Nesse contexto, objetiva-se calcular a dificuldade de plataformas a partir



Figura 5.2: Exemplificação dos tipos de plataforma, em ordem horária: Inferior Direita, Inferior Esquerda, Superior Esquerda e Superior Direita. Estão indicados origens, destinos e caminhos a serem percorridos por setas. Os destinos possuem salvas as informações da sua primeira e última posição.

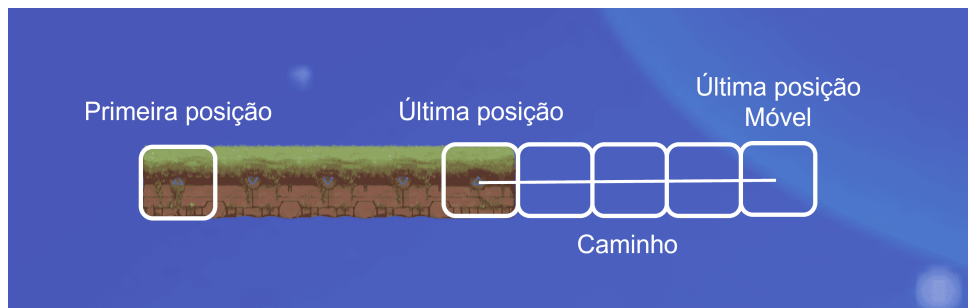


Figura 5.3: Ilustração de plataforma móvel, com informações da primeira e última posição da plataforma e do caminho a ser percorrido. Nota-se que a posição final varia de acordo com o caminho.

da probabilidade de sucesso de um salto de um ponto de origem a um ponto de destino, incluindo novas direções e configurações do salto.

Assim, ao calcular as margens de erro horizontal (m_x) e vertical (m_y), deve-se considerar as Equações 5.1 e 5.2 para os valores de Δx e Δy , respectivamente, assim como representado graficamente na Figura 4.2.

$$\Delta x = \begin{cases} x_2 - x_0, x_2 \leq fim_x \\ \frac{(x_2 + fim_x)}{2} - x_0, x_2 > fim_x \end{cases} \quad (5.1)$$

$$\Delta y = y_1 - y_0 \quad (5.2)$$

Nota-se que a última posição da plataforma, determinada pela distância entre o ponto inicial e seu tamanho (fim_x), precisa ser utilizada na dificuldade do eixo horizontal. Conforme apresentado na Figura 5.4, se a posição final da plataforma for menor do que a distância em que o salto terminaria, deve ser feita a média entre o limite da plataforma e a distância para calcular o erro m_x .

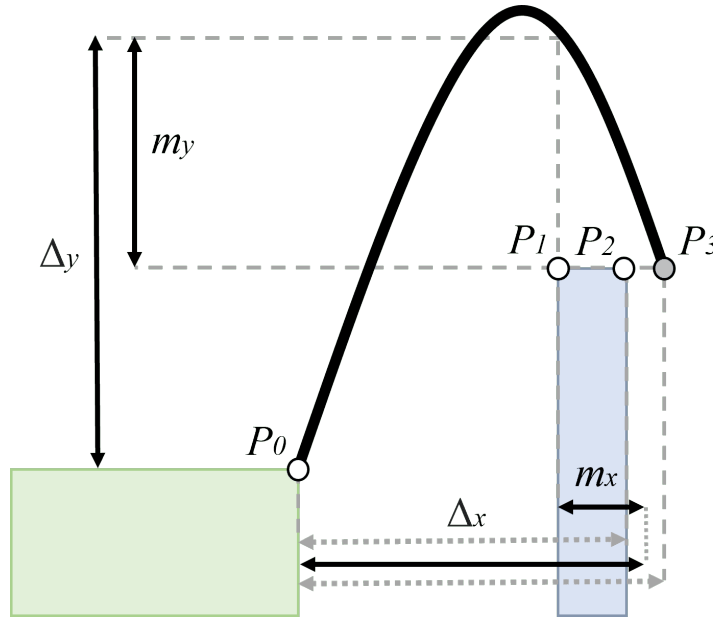


Figura 5.4: Representação gráfica do conceito de margem de erro para saltos em que a posição final da plataforma é menor do que a posição final que o salto terminaria.

Além da definição de protocolos para adaptação de plataformas, os valores de variáveis para as adaptações do salto foram revistos. Considera-se que o jogo pode possuir uma velocidade de impulso do salto e que o avatar pode saltar fora do ângulo de 45° na função quadrática f interseccionando a origem, sendo este ângulo utilizado para cálculo dos valores máximos. Dessa forma, as Equações 4.4, 4.7 e 4.8 foram modificadas a partir das fórmulas da física cinemática para incluir o ângulo de lançamento do salto θ e a posição inicial do avatar nos eixos horizontal e vertical variáveis:

$$f(x) = y_0 + \frac{v_{0y} * \text{sen}\theta}{v_{0x} * \text{cos}\theta} * (x - x_0) - \frac{g}{2 * v_{0x} * \text{cos}\theta^2} * (x - x_0)^2 \quad (5.3)$$

$$y_1 = f(x_1) = y_0 + \frac{v_{0y}}{v_{0x}} * (x_1 - x_0) - \frac{g}{2 * v_{0x}^2} * (x_1 - x_0)^2 \quad (5.4)$$

$$f(y) = x_0 + (v_{0y} * \text{sen}\theta \pm \sqrt{v_{0y}^2 * \text{sen}^2\theta + 2 * g * (y_0 - y)}) * \frac{v_{0x} * \text{cos}\theta}{g} \quad (5.5)$$

$$x_2 = f(y_2) = x_0 + (v_{0y} \pm \sqrt{v_{0y}^2 + 2 * g * (y_0 - y_2)}) * \frac{v_{0x}}{g} \quad (5.6)$$

$$x_r = x_0 + \frac{(v_{0y} + \sqrt{v_{0y}^2 + (2 * g * y_0)}) * v_{0x}}{g} \quad (5.7)$$

$$y_v = \frac{v_{0y}^2 + (2 * g * y_0)}{2 * g} \quad (5.8)$$

Ao ocorrer um impulso, pode-se definir que o avatar teve um ponto de origem deslocado, com x_0 e y_0 possuindo valores. No jogo *The Explorer: 2D*, esse impulso é igual a $x_0 = 3 * v_{0x} * \Delta t$ e $y_0 = 3 * v_{0y} * \Delta t$, sendo Δt o intervalo em segundos em que taxas de quadros fixas do jogo atualizam. Também existe aceleração no eixo x, mas a velocidade horizontal foi tratada como constante pela garantia do avatar conseguir alcançar a velocidade máxima antes do salto iniciar ou durante o período de impulso.

Tendo em vista que o ângulo θ pode ser alterado durante o salto pelo jogador conforme a movimentação no eixo horizontal e que as velocidades dos eixos x e y são independentes, o ângulo que proporciona valores máximos para $f(x)$ e $f(y)$ pode ser utilizado, sendo tratado como um valor constante e desconsiderado para as Equações de y_1 e x_2 . Com isso, o salto também pode ser corrigido pelo jogador para os casos passíveis de correção apresentado pela Figura 4.3 e pelas Equações 4.14 e 4.15, referentes aos valores de c_{min} e c_{max} .

Por sua vez, as fórmulas originais consideram apenas saltos do caso Superior Direita, precisando fazer as seguintes considerações:

- Para os casos do tipo Inferior, o destino fica abaixo da origem e o erro m_y utiliza-se da distância máxima y_v que o jogador consegue percorrer no eixo vertical, como descrito pelas Equações 5.9 e 5.9 e pela Figura 5.5. Ou seja, supõe-se que o jogador tenha a altura máxima de seu salto como fator de dificuldade para locomoção no período de queda, sendo visto como uma dificuldade associada a uma reta ao invés de uma parábola. Caso o destino esteja abaixo da origem a uma distância maior do que a altura do salto, o erro na vertical é tratado como máximo, ainda que o salto seja sempre possível no eixo vertical.

- Para os casos do tipo Esquerda, a movimentação horizontal é feita no sentido contrário com a origem vindo após o destino, mas as Equações utilizadas são as mesmas do tipo Direita, assim como ilustrado na Figura 5.6.

$$m_y = \begin{cases} \Delta y - ini_y = \Delta y - y_2, ini_y \geq y_0 \\ y_v - (y_0 - ini_y) = y_v - (y_0 - y_2), ini_y < y_0 \end{cases} \quad (5.9)$$

$$M_y = \begin{cases} \frac{m_y}{y_v}, ini_y \geq y_0 \\ \frac{y_v}{m_y}, ini_y < y_0 \end{cases} \quad (5.10)$$

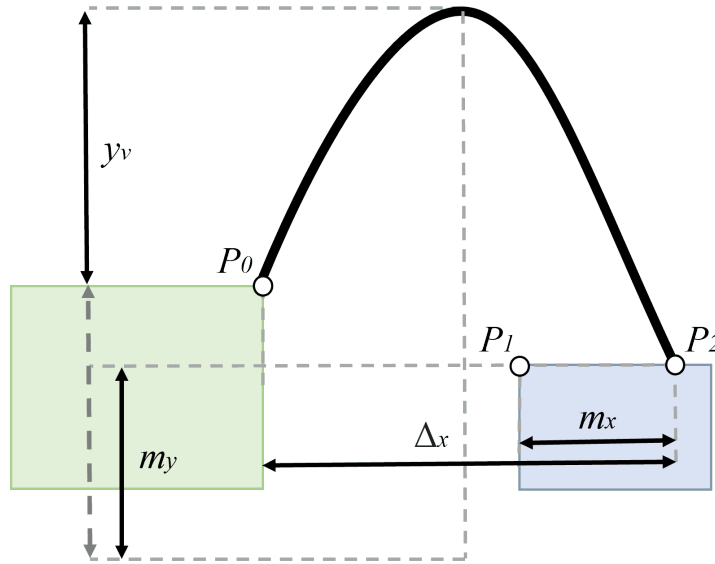


Figura 5.5: Representação gráfica do conceito de margem de erro para saltos em que a origem está acima do destino (Inferior).

Valores constantes foram atualizados para aumentar a amplitude média do salto e possibilitar mais opções de adaptação, em vista da altura e distância máxima do salto impactarem no tamanho dos componentes de plataforma. Com isso, a gravidade g foi alterada de 38 para 36, o valor médio de v_{0y} alterou de 16 para 18 e a v_{0x} manteve com o valor 7. Considera-se que, para adaptação da amplitude do salto, v_{0y} variava na faixa de 14,5 a 18,5, sendo alterado para faixa de 16 a 20. Logo, considerando que a taxa média de atualização de quadros do jogo é 0,02, a altura máxima do salto variava de 3,64 a 5,61 e a distância máxima horizontal de 6,15 a 7,63. Com as alterações de constantes e faixas de valores variáveis, a altura máxima do salto passou a variar de 4,52 a 6,75 e a distância máxima horizontal de 7,04 a 8,60, conforme ilustrado na Figura 5.7.

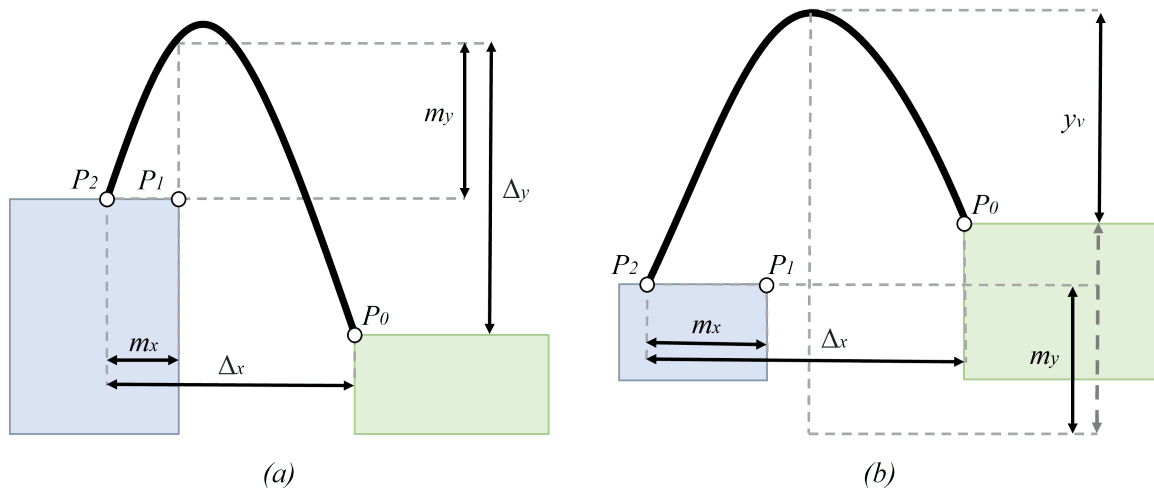


Figura 5.6: Representação gráfica do conceito de margem de erro para saltos em que a origem vem antes do destino (Esquerda), sendo (a) para casos que o destino vem acima da origem (Superior Esquerda) e (b) para casos que o destino vem abaixo da origem (Inferior Esquerda).

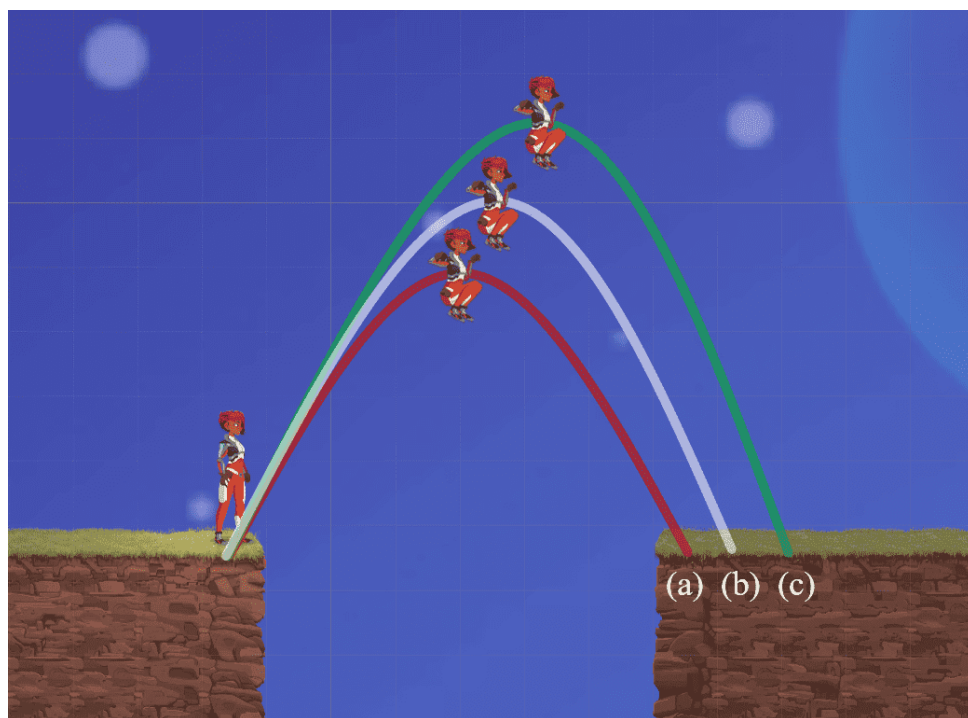


Figura 5.7: Ilustração da diferença de dificuldade para amplitude do salto, sendo que (a) amplitude mínima do salto está representada pela parábola de cor vermelha (mais difícil), (b) média na cor branca e (c) máxima na cor verde (mais fácil).

A dificuldade espacial (D_e), definida entre 0 e 1, é calculada a partir da margem de erro final (M) e, conseqüentemente, das margens de erro normalizadas horizontal (M_x) e vertical (M_y):

$$D_e = 1 - M = 1 - (M_x * M_y) \quad (5.11)$$

Para dificuldade temporal, calcula-se o período de tempo que o salto seria prejudicial ou não para o avatar. A partir disso, é determinada a probabilidade de sucesso ou falha temporal de um salto com t_{np} sendo o tempo de um ciclo do componente que não ocasionará em prejuízo ao jogador e t_p o tempo que ocasionaria prejuízo. No caso de plataformas móveis que não possuem aceleração, observa-se um movimento uniformemente variável e que as distâncias são diretamente proporcionais aos tempos de um ciclo de movimentação da plataforma, sendo D_t a dificuldade temporal, M_t a margem de erro temporal, v_t a velocidade da plataforma, $dist_{np}$ a distância que o salto não será prejudicial ao jogador, $dist_p$ a distância que será prejudicial e M_t a margem de erro temporal do componente:

$$M_t = \frac{t_{np}}{t_p + t_{np}} = \frac{dist_{np}}{(dist_p + dist_{np}) * v_t} \quad (5.12)$$

$$D_t = 1 - M_t \quad (5.13)$$

Caso a origem do salto seja uma plataforma móvel, o cálculo de dificuldade temporal deve ser adicionado ao componente. Nesse cenário, calcula-se a margem de erro temporal M_t a partir do caminho percorrido pela plataforma de origem. Ou seja, correspondendo as janelas de tempo que o salto poderá ser executado com sucesso, verifica-se as distâncias não prejudiciais ($dist_{np}$) e prejudiciais ($dist_p$) ao salto do jogador com a variação do ponto de origem entre a última posição e a última posição móvel do componente de origem, ilustradas na Figura 5.3. Para casos em que os componentes de origem e de destino são móveis, soma-se as distâncias não prejudiciais e prejudiciais, utilizando ambos os componentes como parâmetro para determinação da dificuldade temporal.

Nota-se que a dificuldade é maior temporalmente para casos em que há uma maior proporção de tempo prejudicial a ser percorrida ou que há uma maior velocidade, ocasionando uma menor janela de tempo para superar o desafio. Por sua vez, a dificuldade total (D) é obtida a partir de cada margem de erro quando a entidade possui dificuldade espaço-temporal, sendo $M_{e_{media}}$ a média das margens de erro espaciais para um ciclo de movimentação do componente:

$$D = 1 - (M_{e_{media}} * M_t) \quad (5.14)$$

Baseado nas observações sobre entidades dinâmicas de Mourato [96], considerou-se a independência da probabilidade de sucesso dos estimadores espacial e temporal de cada componente, multiplicando-os para fornecer a dificuldade total do componente. Observa-

se que as dificuldades variam entre 0 e 1 e indicam maior dificuldade quanto maior for o valor.

5.2.3 Atualização do método de medição de dificuldade do nível

O método para medição da dificuldade de um nível foi atualizado para incluir as mudanças da método de cálculo de margens de erro e probabilidade de sucesso de um salto, apresentada na Seção 5.2.1. Além disso, buscou-se abranger a dificuldade para componentes de jogo que não sejam plataformas e buracos, associando componentes genéricos de jogos de plataforma e permitindo que o método seja utilizado em outros subgêneros de plataforma além do clássico. Desse modo, o método consegue calcular a dificuldade total de um nível para qualquer jogo de plataforma, considerando os componentes descritos por Smith [7]:

- Plataformas: calculado a partir das probabilidades de sucesso espacial e temporal, descritas nas Equações 4.13, 5.12 e 5.14.
- Obstáculos (estendendo o conceito para objetos que dificultam o avanço do avatar):

- Inimigos, sendo *inimAlcance* o alcance do ataque e *inimVida* a vida do inimigo:

$$1 - \frac{1}{inimAlcance * inimVida} \quad (5.15)$$

- * O alcance do ataque pode ser considerado como a distância que o inimigo se movimenta ao atacar (em ataques corpo-a-corpo) ou a distância necessária para detecção do avatar (em ataques a distância).
- * Para inimigos não estáticos, deve-se considerar a probabilidade de erro adicionada para entidades dinâmicas. Logo, multiplica-se a dificuldade estática do inimigo pela equação de tempo aplicada também para plataformas, assim como descrito nas Equações 5.12 e 5.14.

- Armadilhas, sendo *armDist* a distância a ser percorrida para evitar a armadilha:

$$1 - \frac{1}{armDist} \quad (5.16)$$

- * Considera armadilhas estáticas, como espinhos, analisando qual a distância deve ser evitada para cada direção e multiplicando os resultados. Caso a armadilha tenha limites horizontais, verifica qual a distância livre após o obstáculo para locomoção do avatar. Caso a armadilha tenha limites verticais, observa qual a distância vertical que o avatar terá para se locomover.

- Objetos destrutíveis, sendo $qtdAcertosObj$ a quantidade de vezes que se precisa acertar o obstáculo para que ele seja destruído:

$$0.5 - (0.5/qtdAcertosObj) \quad (5.17)$$

- Ajudas de movimento:

- Objetos capazes de serem empurrados, sendo $distObj$ a distância que o objeto deve ser empurrado para que alcance seu local de destino:

$$0.5 - (0.5/distObj) \quad (5.18)$$

- * Também são categorizados como plataforma, quando for analisar o salto a ser feito a partir dele.

- Itens colecionáveis: não possuem dificuldade intrínseca, mas associada aos demais componentes para que seja coletado.
- Gatilhos: são dependentes de características específicas do jogo e de seu comportamento.

Observa-se que objetos que não causam morte, como objetos destrutíveis e capazes de serem empurrados, possuem um limite de dificuldade estabelecido (metade do máximo) e que ações triviais como andar e subir uma escada não são relevantes para o cálculo de dificuldade. A partir da modularização da dificuldade dos componentes, outros componentes podem ser incluídos e as equações específicas podem ser revistas de acordo com as especificidades do jogo de plataforma. Por exemplo, outras configurações de inimigos podem ser utilizadas para determinação de dificuldade, sendo utilizados os parâmetros mais comuns entre distintos jogos do gênero para as equações apresentadas acima.

Com a dificuldade de cada componente individual medida, a dificuldade do jogo é calculada a partir da média de dificuldade dos componentes ($diffComp_{media}$) e da quantidade de componentes ($qtdComp$):

$$diff_{nivel} = diffComp_{media} * (1 - baseDif^{-qtdComp}) \quad (5.19)$$

Considera-se $baseDif$ com valor de 1.1 para o contexto do jogo, esse valor pode ser modificado de acordo com a quantidade de níveis e componentes utilizados no jogo. Desse modo, para o jogo em questão, a partir de 50 componentes em um nível, a dificuldade será baseada primordialmente a partir da média de dificuldade dos componentes ao invés da quantidade de componentes presente no nível. Isso se deve aos jogadores demonstrarem

perceber uma dificuldade crescente da dificuldade pela quantidade de componentes, mas de forma limitada e com maior correlação a dificuldade geral dos componentes aplicados. Assim, a quantidade de componentes não é diretamente indicativa da dificuldade do nível, mas um intermédio entre esta e a dificuldade média.

Esse formato de medição integra os estudos de componentes de jogos de plataforma de [7], a definição de dificuldade de plataforma proposta por Mourato [1] e as modificações descritas na Seção 5.2.1 para ampliar os conceitos de salto em múltiplas direções e considerar casos mais amplos de configuração do salto do avatar.

Nota-se que a Equação 5.19 deve ser aplicada em níveis com caminhos lineares. Para cenários exploratórios e de enigma, deve ser utilizado um grafo de caminhos a serem percorridos, definindo a dificuldade mínima e máxima de acordo com os caminhos possíveis de serem percorridos pelo jogador, descrito na Equação 5.20. Nesse contexto, pode-se considerar a média entre os valores mínimo e máximo como a dificuldade estimada para o nível. Ademais, níveis lineares podem incluir esse tipo de métrica para incluir caminhos opcionais, geralmente para que o jogador pegue colecionáveis ou faça atividades adicionais.

$$dif_{nível} = [difComp_{media_{min}} * (1 - 1.1^{-qtdComp_{min}}), difComp_{media_{max}} * (1 - 1.1^{-qtdComp_{max}})] \quad (5.20)$$

Considera-se $difComp_{media_{min}}$ e $qtdComp_{min}$ como a dificuldade média e a quantidade de componentes no menor caminho para alcançar o final do nível, ignorando caminhos opcionais e utilizando o caminho de menor dificuldade quando existir opções paralelas a se percorrer. Por sua vez, considera-se $difComp_{media_{max}}$ e $qtdComp_{max}$ como a dificuldade média e a quantidade de componentes no maior caminho para alcançar o final do nível, incluindo todos os caminhos opcionais e utilizando o caminho de maior dificuldade quando existir opções paralelas a se percorrer.

5.2.4 Modificações do Sistema de Análise

Considerando os níveis elaborados para novas experimentos, nota-se que níveis de ADD devem ter buracos e plataformas com distância máxima entre o ponto de origem e o ponto de destino correspondente a distância máxima a ser percorrida em um salto, nos eixos horizontal e vertical. Por sua vez, caso ocorra adaptação de salto no experimento, deve-se aplicar as distâncias máximas de buracos e plataformas conforme a velocidade mínima possível para v_{oy} , garantindo que todos os saltos serão viáveis.

Além disso, tendo em vista que caminhos percorridos após uma morte correspondem a uma maior dificuldade de um segmento, saltos de sucesso são desconsiderados para os

parâmetros de entrada de heurísticas em trechos que são percorridos novamente, desde que o caminho seja linear. Quando o nível é exploratório, a categorização entre saltos de sucesso e falhos deve ser mantida normalmente pelo jogador poder executar ações pertinentes ao nível que não sejam somente alcançar o final, como resolver um enigma ou pegar colecionáveis.

As heurísticas foram redefinidas para serem únicas independente da adaptação a ser feita no Sistema de Controle e para considerarem os parâmetros identificados como de maior correlação a dificuldade percebida pelo jogador, com as respectivas observações a serem apresentadas na Seção 6.2. Com as novas heurísticas para o Sistema de Análise, objetivou-se reduzir o uso de constantes fixas, sendo utilizado base exponenciais proporcionais (*base* e *baseInferior*) ao número de níveis presentes no jogo (*qtdNiveis*) e que consideram a expectativa de dificuldade crescente entre níveis:

$$base = 1 + (1/qtdNiveis) \quad (5.21)$$

$$baseInferior = 1 + (1/qtdNiveis^2) \quad (5.22)$$

Considera-se *qtdNiveis* como a quantidade de níveis da sessão de jogo e que a *base* é utilizada como base de expoentes que variam entre 0 e um valor delimitado, como 1 ou a quantidade de níveis, e a *baseInferior* como base de expoentes que começam com 0 e não possuem limite superior ou em caso de morte. Com isso, ajustes ao morrer são mais suavizados em relação a quando é gerado um novo nível.

Uma problemática de análise dos experimentos de Santos e Moraes que comparavam adaptação de plataformas e do salto se refere a existirem diferenças no Sistema de Análise, enquanto somente o Sistema de Controle estava sendo comparado. Com isso, considera-se que os resultados encontrados eram influenciados não só pelas diferenças entre os modelos de adaptação, mas também pela forma que eram calculados.

Para novos experimentos, as heurísticas apresentadas na Subseção 4.4.2 foram revistas através da padronização do Sistema de Análise quando comparando modelos distintos do Sistema de Controle. Considerando esse escopo, os seguintes parâmetros de entrada e heurísticas correspondentes foram incluídos:

- Quantidade de mortes:
 - Ao morrer, sempre facilita com

$$(1 - baseInferior^{-qtdMortes}) \quad (5.23)$$

- Ao criar nível, facilita se houve mortes no nível anterior e dificulta se não houve mortes no nível anterior:

* Ao facilitar, diminui

$$(1 - baseInferior^{-qtdMortesAnt}) \quad (5.24)$$

* Ao dificultar, aumenta

$$(1 - base^{-qtdSemMorte}) \quad (5.25)$$

– Considera-se $qtdMortes$ como a quantidade de mortes no nível atual, $qtdMortesAnt$ como a quantidade de mortes no nível anterior e $qtdSemMorte$ como a quantidade de níveis consecutivos sem morte.

• Quantidade de saltos de sucesso:

– Ao morrer, facilita se ocorreram mais saltos falhos do que de sucesso e dificulta se ocorreram mais saltos de sucesso do que falhos desde a última morte ou do início do nível até a morte atual:

* Ao facilitar, diminui

$$1 - baseInferior^{saltoSucessoMorte/(saltoSucessoMorte+saltoFalhoMorte)} \quad (5.26)$$

* Ao dificultar, aumenta

$$1 - baseInferior^{saltoFalhoMorte/(saltoSucessoMorte+saltoFalhoMorte)} \quad (5.27)$$

– Ao criar nível, facilita se ocorreram mais saltos falhos do que de sucesso e dificulta se ocorreram mais saltos de sucesso do que falhos no nível anterior:

* Ao facilitar, diminui

$$1 - base^{saltoSucesso/(saltoSucesso+saltoFalho)} \quad (5.28)$$

* Ao dificultar, aumenta

$$1 - base^{saltoFalho/(saltoSucesso+saltoFalho)} \quad (5.29)$$

– Considera-se $saltoSucessoMorte$ como a quantidade de saltos de sucesso desde a última morte ou do início do nível, $saltoFalhoMorte$ como a quantidade de saltos falhos desde a última morte ou do início do nível, $saltoSucesso$ como a quantidade de saltos de sucesso no nível anterior e $saltoFalho$ como a quantidade de saltos falhos no nível anterior.

- Tempo jogado:

- Ao morrer, facilita se o tempo jogado for maior do que a distância a ser percorrida até o fim do segmento atual:

- * Ao facilitar, diminui

$$1 - baseInferior^{1-(distancia/tempo)} \quad (5.30)$$

- Ao criar nível, facilita se o tempo jogado for maior do que a distância total do nível e dificulta se o tempo jogado for menor do que a distância total do nível:

- * Ao facilitar, diminui

$$1 - base^{1-(distanciaNivel/tempoNivel)} \quad (5.31)$$

- * Ao dificultar, aumenta

$$1 - baseInferior^{distanciaNivel/tempoNivel} \quad (5.32)$$

- Considera-se *distancia* como a distância a ser percorrida desde o início do nível até o segmento atual, *tempo* como o tempo jogado no nível, *distanciaNivel* como a distância total do nível anterior e *tempoNivel* como o tempo jogado no nível anterior.

Nas fórmulas apresentadas acima, a distância a ser percorrida corresponde a soma da distância nos eixos x e y entre o ponto inicial, no qual o avatar inicia o nível, e o ponto final do segmento (*checkpoint*) ou do nível (portal). Além disso, a variável de dificuldade global só aumenta ao criar um nível caso o anterior tenha sido completado, evitando dificultar em caso de desistência.

Assim como em experimentos anteriores, a dificuldade global (*dif*) aumenta e diminui de acordo com as heurísticas definidas por dados de desempenho e é limitada conforme o nível atual jogado para permitir progressão de dificuldade no jogo, assim como visto nas Equações 4.16 e 4.17. O valor da dificuldade global foi padronizado como único para novos experimentos, independente do modelo testado, incluindo as mesmas heurísticas e sempre iniciando com 0,25, buscando induzir a progressão gradativa da dificuldade durante a sessão de jogo. Ao iniciar com o valor intermediário, o jogador poderia alcançar a dificuldade máxima antes dos níveis finais.

Observa-se que os ajustes são aplicados ao gerar segmentos de um nível ou quando o avatar morre, não sendo feito em tempo real nos experimentos. Isso evita a percepção direta das adaptações pelo usuário enquanto joga e que ocorra erros de interação com o

usuário. Ademais, as mortes consideradas para as heurísticas dos experimentos anteriores consideravam somente quedas em abismos ou ácidos, enquanto no novo formato consideraram os demais tipos de mortes, como por inimigos ou armadilhas. Essa mudança foi feita para adaptação, mesmo com foco em plataformas ou salto, considerar que o contexto global de dificuldade do nível impacta na experiência do usuário, não somente os componentes diretamente relacionados a adaptação.

Para os novos experimentos, foi feita uma simplificação dos casos de adaptação para que ocorram com maior frequência. Anteriormente, as adaptações ocorriam proporcionalmente a dificuldade média de todos os componentes do nível, de acordo com $diffCompModif$ das Equações 4.28 e 4.29. No formato atualizado, as adaptações são feitas de acordo com dificuldades individuais dos componentes. Logo, a dificuldade de cada componente pode mudar caso a diferença entre a dificuldade global e a dificuldade do componente ($auxDiffComp$, descrita na Equação 4.27) seja menor ou maior do que os limiares definidos nas Equações 4.25 e 4.26, sendo que os limiares indicam se a diferença de dificuldade terá algum impacto no eixo x ou y do componente:

$$diffCompModif_x = auxDiffComp, |auxDiffComp| > limiar_x \quad (5.33)$$

$$diffCompModif_y = auxDiffComp, |auxDiffComp| > limiar_y \quad (5.34)$$

Dessa forma, cada componente pode variar de forma independente para alcançar a dificuldade desejada para o nível separadamente nos eixos x e y, mantendo um padrão de *design* no qual os componentes são adaptados sem a interferência da dificuldade de outros componentes do nível e suavizando as adaptações que ocorriam em conjunto para múltiplos componentes ao passar de limiares do nível. Por sua vez, para adaptação da amplitude do salto, a diferença é sempre usada diretamente na variável de velocidade do avatar no eixo y:

$$diffModif_{salto} = auxDiffComp \quad (5.35)$$

Ou seja, a variável v_{o_y} é decrementada ou incrementada na Equação 4.4, considerando que a gravidade e a velocidade horizontal são constantes. Ademais, para novos experimentos, foi definido que a adaptação do salto deve ser feita antes da adaptação de plataformas e que os limiares ($limiar_x$ e $limiar_y$) devem ser recalculados, pois a distância máxima percorrida em um salto horizontalmente (x_r) e verticalmente (y_v) altera com a mudança da variável v_{o_y} , conforme as Equações 4.9 e 4.10.

O dado extraído a partir de $diffCompModif$ indica quanto (valor) e como (sentido) o componente deve ser alterado, sendo o parâmetro de saída do Sistema de Análise a ser

utilizada individualmente pelos componentes no Sistema de Controle, de acordo com as seguintes características:

- Adaptação do tamanho de plataformas (nos eixos x e y):
 - Facilita ao diminuir distância horizontal e/ou vertical entre plataformas (valores negativos de $diffCompModif_x$ e $diffCompModif_y$, respectivamente).
 - Dificulta ao aumentar distância horizontal e/ou vertical entre plataformas (valores positivos de $diffCompModif_x$ e $diffCompModif_y$, respectivamente).
- Adaptação da amplitude do salto:
 - Facilita ao diminuir a amplitude do salto (valores negativos de $diffModif_salto$).
 - Dificulta ao aumentar a amplitude do salto (valores positivos de $diffModif_salto$).

5.3 Modelo de Aprendizado de Máquina

As técnicas de Aprendizado de Máquina são aplicadas em uma ampla gama de finalidades no campo de jogos eletrônicos. Os estudos de Ajuste Dinâmico de Dificuldade são crescentes nesse âmbito, porém, poucos estudos na literatura combinam o Aprendizado de Máquina e características intrínsecas ao jogador, como feito por Hawkins [11] para identificar aversão ao risco por meio de um filtro de partículas.

Ou seja, além da medição da dificuldade experienciada, os jogadores podem ser categorizados em diferentes grupos. Essas categorias possibilitam a adaptação do conteúdo dos níveis com base nos perfis de jogadores, aprimorando a lógica quando comparada às adaptações baseadas exclusivamente no desempenho do jogador.

Em vez de usar os dados brutos do jogador para adaptar o conteúdo do jogo por meio de heurísticas, eles podem ser usados para categorizar o jogador dentro dos padrões de perfil de jogador e níveis de habilidade. A lógica básica para o equilíbrio da dificuldade do jogo considera que a dificuldade experienciada pelo jogador deve ser suficientemente desafiadora para manter o interesse no jogo. No entanto, os jogadores podem se divertir com outros tipos de recompensa além da dificuldade linear do jogo. Por exemplo, um jogador pode preferir uma experiência mais focada em exploração ou na conclusão rápida do jogo.

O reconhecimento dos perfis de jogador deve ser preciso e eficiente para a adaptação do jogo. Esta abordagem considera o reconhecimento automático de perfis durante o jogo, inferindo a partir de dados brutos, como interações e resultados alcançados. Além disso, a classificação automática das métricas do jogo e os recursos relevantes para traços comportamentais específicos devem ser destacados para a aplicação deste modelo.

5.3.1 Adaptação por dificuldade experienciada e perfil de jogador

O presente trabalho utiliza-se de método de aprendizagem supervisionada para classificação, tendo como saída a dificuldade experienciada e o perfil do jogador, conforme o modelo Hexad [3]. Esse modelo pode ser aplicado independente do gênero do jogo e possui associação direta de traços de personalidade com características de *design* de componentes de jogo. Ademais, os questionários foram validados, inclusive para português [106], e possuem vocabulário de fácil entendimento para usuários que não são familiares com o *design* de jogos [57].

Após coletar dados suficientes e calcular pesos e vieses, uma Rede Neural Artificial (RNA) direta foi codificada no contexto do jogo (C# para Unity) de forma a tomar decisões desejadas para adaptação a partir de atributos atuais da sessão de jogo. Nesse cenário, foram utilizados dados capturados em um experimento que media, por meio de questionário, a dificuldade de cada nível (em uma escala de 0 a 10) e os perfis do jogador.

Para o contexto de jogo de plataforma, foram desconsiderados os perfis Socializador e Filantropo por estarem associados a características interpessoais e o estudo tratar apenas jogos para um único jogador. Os demais perfis foram identificados em uma escala de pertencimento de 0 a 1 para cada jogador, definidos conforme o modelo de questionário [105].

A primeira etapa para construção da RNA foi a seleção de atributos úteis para o banco de testes. Assim, foram identificados os parâmetros com maior correlação a dificuldade indicada pelo jogador no nível e, separadamente, ao seu perfil, sendo que as observações para essas escolhas estão apresentadas na Seção 6.2. Os seguintes parâmetros foram definidos:

- Dificuldade:
 - Entradas:
 - * Quantidade de mortes até a última morte ou durante o nível.
 - * Porcentagem de saltos de sucesso em relação ao total de saltos falhos e de sucesso até a última morte ou durante o nível.
 - * Tempo jogado até a última morte ou durante o nível.
 - Saída: dificuldade experienciada pelo jogador.
- Perfil do jogador:
 - Entradas:

- * Porcentagem de colecionáveis coletados em relação aos colecionáveis disponíveis no nível.
 - * Porcentagem de ataques corpo-a-corpo em relação ao total de ataques no nível, considerando corpo-a-corpo e a distância.
 - * Porcentagem de tempo andando em direção ao ponto final do nível em relação ao total tempo total andando no nível. Observa-se que, geralmente, o ponto final está a direita do ponto inicial do nível.
- Saída: diferença entre o pertencimento nos perfis de Realizador e Espírito Livre.

Para seleção de atributos, as entradas da RNA estão em conformidade com as entradas de heurísticas, as quais também buscam adaptar o jogo pela dificuldade experienciada pelo jogador, descrito na Seção 5.2.4. Quanto maior é a quantidade de mortes e o tempo jogado e quanto menor é a porcentagem de saltos de sucesso, estima-se uma maior dificuldade experienciada no nível.

Por sua vez, para definição de perfil do jogador, os atributos consideram características de exploração e métodos de ação dos jogadores. Quanto maior a quantidade de colecionáveis coletados e o tempo andando na direção oposta ao ponto final e quanto menor o uso proporcional de ataques corpo-a-corpo, estima-se uma maior busca pela maestria do nível, com o usuário estando mais próximo do perfil de Realizador do que do perfil de Espírito Livre. Nesse contexto, usuários que buscam maestria aprendem a utilizar melhor os controles do avatar, como eliminar inimigos antes de entrar em contato com eles, e executam mais atividades disponíveis no nível além do objetivo primário de alcançar o final.

Somente foram considerados os perfis de Realizador e Espírito Livre por serem características intrínsecas ao jogador, diferentemente dos perfis de Jogador e Revolucionário que dependem de características extrínsecas, como recompensas ou mudanças do sistema do jogo. Enquanto jogadores de Espírito Livre preferem autonomia, exploração e uma diversão mais fácil, Realizadores buscam maestria, desafios e uma diversão mais difícil [57]. Além disso, considera-se a diferença de valores de pertencimento entre esses perfis para RNA, simplificando em uma única saída.

Observa-se que um Realizador poderia demorar mais para completar o nível ou ter mais mortes ao buscar eliminar todos os inimigos ou pegar todos os colecionáveis, ainda que tenha um alto nível de habilidade e não considere o nível difícil. Esse tipo de situação demonstra a importância de utilizar informações como o perfil do jogador para adaptação da dificuldade, considerando que a medição direta da dificuldade por dados da sessão de jogo não considera um escopo mais completo das características do jogador.

Em ambos os casos (dificuldade experienciada e perfil de jogador), temos 3 atributos na camada de entrada e 1 atributo na camada de saída, com 1 camada oculta com 2 neurônios entre as camadas de entrada e saída para processamento dos dados de entrada, ilustrado na Figura 5.8. A regra geral utilizada foi de que o número de neurônios ocultos deve estar entre o tamanho das camadas de entrada e saída [110].

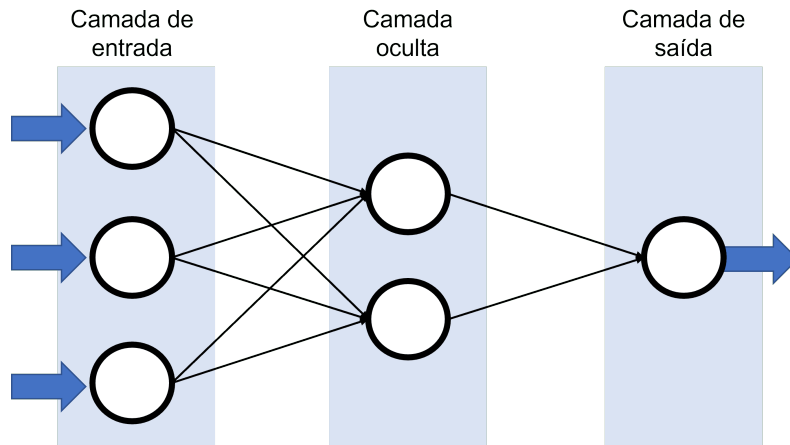


Figura 5.8: Exemplificação de Rede Neural Artificial.

O treinamento da RNA serviu para calcular os pesos e vieses, de forma a que pudesse ser utilizada posteriormente em sessões do jogo, sendo feito com o algoritmo de retropropagação. Definiu-se que os valores dos atributos deveriam estar todos entre 0 e 1 para entradas e saídas, sendo também arredondados para duas casas decimais e normalizados, caso necessário. Ademais, foi utilizada a função de ativação sigmoide, considerando a escala dos dados entre 0 e 1.

Os treinamentos rodaram 1000 vezes com a taxa de aprendizado iniciando com 0,8 e finalizando com 0,3 em ambos os casos, definindo quanto qualquer amostra de dados do treinamento afeta a rede como um todo. A taxa de aprendizado era atualizada, indicando se deveria aumentar ou diminuir, conforme a soma dos erros quadrados. Esta é uma ferramenta estatística para determinar quão próximo o modelo construído se encaixa nos dados, buscando assim um valor menor quando a RNA se aproxima de representar o estado atual dos dados. Ou seja, era definida a soma dos erros quadrados da diferença entre as saídas desejadas e calculadas, comparando o valor atual e o anterior e almejando o menor valor, sendo menor do que 0,001 para ambas as redes neurais treinadas.

Após o treinamento, os pesos e vieses obtidos foram usados para a computação de valores estimados referentes a dificuldade experienciada e perfil de jogador. Assim como no caso do uso de heurísticas, a dificuldade do nível é adaptada quando o jogador morre e ao criar o nível. No primeiro caso, é vista a diferença entre a dificuldade do nível (*diffNivel*)

e a dificuldade estimada pela RNA após uma morte ($diffRNA_{morte}$) para determinar o quanto deve ser adaptado no nível em cada morte ($diffModif_{morte}$):

$$diffModif_{morte} = -(diffRNA_{morte} * (1/qtdNiveis^2)) \quad (5.36)$$

Ao morrer, a dificuldade mantém ou diminui, nunca aumentando, em vista de mortes indicarem que não está sendo fácil para o usuário completar o nível. A mudança é feita em uma escala proporcional a quantidade de níveis da sessão de jogo ($qtdNiveis$). Nota-se que, a cada morte, a dificuldade experienciada seja estimada com um valor maior, causando a diminuição gradativa da dificuldade do nível.

Ao criar um nível, segue-se o mesmo padrão para adaptação, verificando quanto a dificuldade deve ser modificada para o nível a ser criado ($diffModif_{nivel}$), aumentando a dificuldade caso a do nível anterior seja maior do que a estimada pelo RNA para o mesmo nível ($diffRNA_{nivel}$). Ou seja, se os dados do jogador indicam que sua experiência foi correspondente a um nível mais difícil do que o jogado, o próximo precisa ser mais fácil. Por outro lado, se os dados do jogador indicam que sua experiência foi correspondente a um nível mais fácil do que o jogado, o próximo precisa ser mais difícil, descrito na Equação:

$$diffModif_{nivel} = \begin{cases} -(diffRNA_{nivel} * (1/qtdNiveis)), & diffNivel < diffRNA_{nivel} \\ (1 - diffRNA_{nivel}) * (1/qtdNiveis), & diffNivel > diffRNA_{nivel} \end{cases} \quad (5.37)$$

Em contrapartida, a estimação do perfil do jogador pela RNA ($diffRNA_{perfil}$) serve para identificar o quanto o jogador se aproxima mais do perfil de Realizador do que do perfil de Espírito Livre, devendo possuir um maior desafio quanto maior for o valor (isto é, quanto mais próximo estiver do perfil de Realizador). Nesse sentido, a dificuldade do nível a ser criado ($diffNivel_{atual}$) cresce gradativamente em conformidade ao tipo de desafio que o perfil do jogador se associa, sendo mais difícil para Realizador e mais fácil para Espírito Livre:

$$diffModif_{perfil} = diffRNA_{perfil} * (1/qtdNiveis) \quad (5.38)$$

$$diffNivel_{atual} = diffNivel + diffModif_{perfil} + diffModif_{nivel} \quad (5.39)$$

Ou seja, a classificação do jogador em um perfil entre Espírito Livre e Realizador é utilizada para determinar a progressão da dificuldade no jogo, em complemento aos ajustes feitos a partir da dificuldade experienciada em cada nível.

5.4 Modificações adicionais do jogo

O jogo *The Explorer: 2D* foi adaptado conforme descrito na Seção 4.5.3. Todavia, novas alterações foram feitas para adequação às propostas deste trabalho. Primeiramente, o jogo foi modificado para permitir jogar online em um website acessível por computadores Windows, Linux e macOS que suportem WebGL ou em um aplicativo Android, exemplificado na Figura 5.9.



Figura 5.9: Captura de tela da interface do jogo em versão de aplicativo Android.

Foi necessário adaptar configurações do jogo em Unity para habilitar o funcionamento em ambas as plataformas e conectar o jogo em uma base de dados, salvando os dados coletados durante a sessão de jogo no banco de dados NoSQL Firebase Cloud Firestore. Adicionalmente, foi incluído no início do jogo o termo de consentimento e orientações para participação na pesquisa.

Também foi criado um nível de tutorial, ilustrado na Figura 5.10, para ser jogado no início de cada sessão de jogo. No tutorial são apresentados todos os componentes, os controles do jogo e orientações gerais. Ou seja, explica como interagir com cada tipo de componente, as formas que se pode morrer no jogo, a característica do jogo ter vidas infinitas nos experimentos, o objetivo de alcançar o final de cada nível com o avatar e a opção de rever controles e desistir de um nível no menu de pausa.

Observa-se que o tutorial serve como base de referência para as adaptações nos níveis seguintes, não tendo nenhum ajuste aplicado nele independente do modelo testado. Além disso, o tutorial é importante para que as experiências dos jogadores não seja influenciada pelo processo de aprendizado de controles e de primeiras interações com distintos componentes.



Figura 5.10: Captura de tela do nível de tutorial.

Algumas mudanças pontuais foram executadas para melhoria da experiência do jogador e para evitar diferença de experiência motivada por características específicas de configurações do jogo:

- A câmera do jogo foi modificada para ter maior escopo de visão do mapa do nível, especialmente considerando que o jogo seria aplicado em novas formatos de tela.
- A movimentação durante ataque corpo-a-corpo do avatar foi bloqueada, pois possibilitava aumentar a amplitude do salto.
- A opção de retornar para o *checkpoint* foi incluída no menu de pausa, considerando casos que o jogador poderia estar com alguma dificuldade para avançar e não estar sendo diretamente monitorado como em testes anteriores.
- As proporções de textos e elementos de interface do jogo foram modificadas para facilitar a visualização em aparelhos menores, como celulares.
- Interface de controle incluída para aplicativo, considerando escopo em que não existe a opção de teclado e mouse.
- Fundos padrões dos níveis que não interferissem visualmente nos componentes, mantendo uma consistência visual e possibilitando melhoria de performance do jogo.

A conexão com o banco de dados e as estruturas do jogo foram programados para mitigar e tratar falhas de conexão e de processamento em distintas máquinas, incluindo

o carregamento do nível por completo antes de iniciá-lo e a otimização dos componentes do jogo. Nesse sentido, também foram feitas adaptações visuais e sonoras conforme recomendações de participantes de experimentos anteriores, como emissão sonora ao coletar itens colecionáveis e indicativo visual ao salvar progresso em *checkpoint*.

Capítulo 6

Testes e Resultados

Neste capítulo serão descritos os testes executados com participantes voluntários do presente estudo, incluindo as análises e os resultados obtidos a partir deles.

6.1 Testes

Este trabalho objetiva investigar a robustez de métricas de criação de níveis e medição do estado de fluxo e da adequação da dificuldade, analisar as atualizações do método de estimativa de dificuldade de níveis de jogos de plataforma, averiguar a categorização de jogadores pela dificuldade experienciada e por perfis a partir de dados brutos da sessão de jogo e comparar a eficiência de distintos modelos de Ajuste Dinâmico de Dificuldade nos Sistemas de Controle e Análise. Nesse contexto, decidiu-se executar 3 baterias com grupos de amostra a partir do jogo *The Explorer: 2D*.

Na primeira bateria, o jogo conteve 8 níveis testando diferentes componentes e situações de dificuldade para jogos de plataforma. Desse modo, pretendia-se verificar a associação entre componentes e o método de estimação de dificuldade, proposto nas Subseções 5.2.2 e 5.2.3.

56 voluntários participaram, com 51 considerados para análise após exclusão de *outliers* que responderam questionários indevidamente. Dentre esses participantes, 35 completaram todos os níveis da sessão de jogo. As idades variaram de 15 a 55 anos, com média de 26 anos, e houve uma maior participação masculina (78%) em comparação a feminina (22%). Além disso, houve uma maior incidência de jogadores com predileção pela dificuldade média (61%) em comparação a difícil (24%) e fácil (15%).

Em uma escala de 1 a 7, os jogadores estavam familiarizados com jogos de plataforma (média 5 e mediana 6) e se consideravam entre um intermediário de casual e dedicado (média 4 e mediana 5). Também se constatou que os participantes jogam com maior

frequência em computadores (média 4 e mediana 5), seguido de dispositivos móveis (média 3 e mediana 4) e outros meios, como consoles (média 2 e mediana 2).

Observa-se que 61% dos participantes jogaram pelo computador e os demais em aparelhos Android. Por sua vez, dentre os perfis Hexad medidos, houve maior ocorrência de jogadores com predominância do perfil Espírito Livre (37%) em comparação a Jogador (31%), Realizador (24%) e Revolucionário (8%).

O algoritmo de estimação da dificuldade já havia sido testado por Rosa [13], assim como detalhado nas Subseções 3.5.2 e 4.6.1. Todavia, como descrito na Subseção 5.2.3, alterações foram feitas para incluir a estimação de dificuldade de mais opções de salto envolvendo plataformas e de outros componentes do gênero, como inimigos e objetos de auxílio. Nesse contexto, decidiu-se identificar a relação entre os dados referentes a sessão de jogo dos participantes e a dificuldade experienciada ou os perfis de jogador Hexad [3]. Assim, as seguintes Hipóteses foram verificadas:

Hipótese 4 *A dificuldade experienciada pelos jogadores pode ser estimada pelos componentes de um nível.*

Hipótese 5 *Os perfis de jogador se relacionam significativamente a dados de desempenho específicos dos jogadores.*

Hipótese 6 *A dificuldade dos níveis se relaciona significativamente a dados de desempenho específicos dos jogadores.*

Na segunda bateria, o jogo apresentou 10 níveis com extremos de dificuldade. Ou seja, ao apresentar níveis alternando entre fáceis e difíceis, esperava-se evitar a co-variância entre a progressão de dificuldade do *design* dos níveis e as adaptações executadas pelos modelos de ADD no Sistema de Controle. Essa bateria serviu para testar as mudanças no Sistema de Controle e Análise do ADD, descritas nas Seções 5.2.1 e 5.2.4. Também teve como objetivo revisar os testes de Santos e Moraes [30] por executarem distintas análises dos dados brutos do jogo para cada tipo de adaptação. Essa revisão se torna necessária para validar instrumentos de medição do estado de fluxo e para comparar adaptações do Sistema de Controle sem variações do Sistema de Análise.

Quatro modelos de adaptação no Sistema de Controle foram comparados: (1) sem a aplicação de ADD, (2) com adaptação do tamanho de plataformas, (3) com adaptação da amplitude do salto e (4) com adaptação combinada do tamanho das plataformas e da amplitude do salto. Nota-se que 126 voluntários participaram, com 112 considerados para análise após exclusão de *outliers* que responderam questionários indevidamente ou ficaram períodos maiores do que 10 minutos estáticos durante o jogo, demonstrando desistência. Observa-se uma divisão igualitária de 28 participantes para cada modelo testado (sem

ADD, ADD de plataforma, ADD de salto e ADD combinado). Dentre esses participantes, 72 completaram todos os níveis da sessão de jogo.

As idades variaram de 15 a 55 anos, com média de 25 anos, e houve uma maior participação masculina (82%) em comparação a feminina (18%). Ocorreu também uma maior incidência de jogadores com predileção pela dificuldade média (59%) em comparação a difícil (22%) e fácil (19%). Os jogadores estavam familiarizados com jogos de plataforma (média 5 e mediana 6) e se consideravam entre um intermediário de casual e dedicado (média 4 e mediana 5). Também se constatou que os participantes jogam com maior frequência em computadores (média 4 e mediana 4), seguido de dispositivos móveis (média 3 e mediana 4) e outros meios, como consoles (média 2 e mediana 1).

Ademais, 77% dos participantes jogaram pelo computador e os demais em aparelhos Android. Dentre os perfis Hexad medidos, houve maior ocorrência de jogadores com predominância do perfil Jogador (37%) em comparação a Espírito Livre (32%), Realizador (23%) e Revolucionário (8%).

Nesse contexto, as Hipóteses 2 e 3 foram verificadas para os modelos propostos, identificando se o estado de fluxo medido por meio de questionário possui resultados significativos ao apresentar variabilidade nos conteúdos do jogo.

Na terceira bateria, o jogo apresentou 8 níveis com extremos de dificuldade, adaptados para ter variabilidade em relação ao quinto teste e aplicando recomendações dos participantes prévios. Três modelos do Sistema de Análise foram comparados: (1) sem a aplicação de ADD, (2) com análise baseada em heurísticas e (3) com análise baseada em aprendizado de máquina. No cenário de aprendizado de máquina, foram utilizadas as estimativas de dificuldade experienciada e do perfil do jogador por redes neurais diretas para decisão das dificuldades a serem adaptadas nos níveis e componentes específicos. A progressão de dificuldade foi feita a partir da estimação de pertencimento do jogador nos perfis Realizador ou Espírito Livre do modelo Hexad [3], como descrito na Seção 5.3.

80 voluntários participaram, sendo que 69 foram considerados para análise após exclusão de *outliers* que responderam questionários indevidamente. Observa-se uma divisão igualitária de 23 participantes para cada modelo testado (sem ADD, ADD baseado em heurísticas e ADD baseado em Rede Neural Artificial (RNA)). Dentre esses participantes, 51 completaram todos os níveis da sessão de jogo. As idades variaram de 18 a 45 anos, com média de 25 anos, e ocorreu uma maior participação masculina (65%) em comparação a feminina (35%). Além disso, houve uma maior incidência de jogadores com predileção pela dificuldade média (45%) em comparação a difícil (29%) e fácil (26%).

Os jogadores estavam familiarizados com jogos de plataforma (média 5 e mediana 6) e se consideravam entre um intermediário de casual e dedicado (média 5 e mediana 6). Também se constatou que os participantes jogam com uma frequência em computadores,

dispositivos móveis e outros meios (média 3 e mediana 3 em todos os casos).

Além disso, 62% dos participantes jogaram pelo computador e os demais em aparelhos Android. Dentre os perfis Hexad medidos, houve maior ocorrência de jogadores com predominância do perfil Jogador (37%) em comparação a Espírito Livre (32%), Realizador (27%) e Revolucionário (4%).

Observa-se que somente um modelo no Sistema de Controle foi aplicado, sendo o ADD combinado de adaptação do tamanho de plataformas e amplitude do salto. Desse modo, as seguintes Hipóteses foram definidas:

Hipótese 7 *Um sistema ADD com aprendizado de máquina que define as adaptações pela dificuldade experienciada e pelo perfil do jogador fornece uma dificuldade adequada para o jogador.*

Hipótese 8 *Um sistema ADD com aprendizado de máquina que define as adaptações pela dificuldade experienciada e pelo perfil do jogador mantém o jogador em um estado de fluxo.*

6.1.1 Descrição dos níveis

Na primeira bateria, os níveis foram formulados para testar diferentes combinações de componentes genéricos de jogos de plataforma:

1. Básico (componentes simples, como colecionáveis e plataformas com dificuldade baixa);
2. Nível de plataforma (somente com componentes de plataforma de dificuldade baixa ou média);
3. Exploração (caminhos opcionais com alta densidade de componentes de baixa dificuldade, contendo plataformas, colecionáveis, obstáculos e inimigos);
4. Quebra-cabeça (dificuldade baseada em desafios feitos com plataformas, obstáculos e inimigos);
5. Ação (dificuldade baseada no uso de inimigos e, parcialmente, plataformas de dificuldade baixa ou média);
6. Nível de análise de perfil de risco do jogador (dependente de dificuldades temporais com plataformas móveis);
7. Baixa densidade de componentes de alta dificuldade (contendo plataformas, colecionáveis, obstáculos e inimigos);

- Alta densidade de componentes de média dificuldade (contendo plataformas, colecionáveis, obstáculos e inimigos);

No quarto nível com desafio por quebra-cabeça, era necessário que o usuário empilhasse objetos para alcançar uma plataforma em posição alta, ilustrado na Figura 6.1. Os demais níveis possuem desafios focados no uso de seus componentes, como exemplificado pela Figura 6.2 para o nível 7.

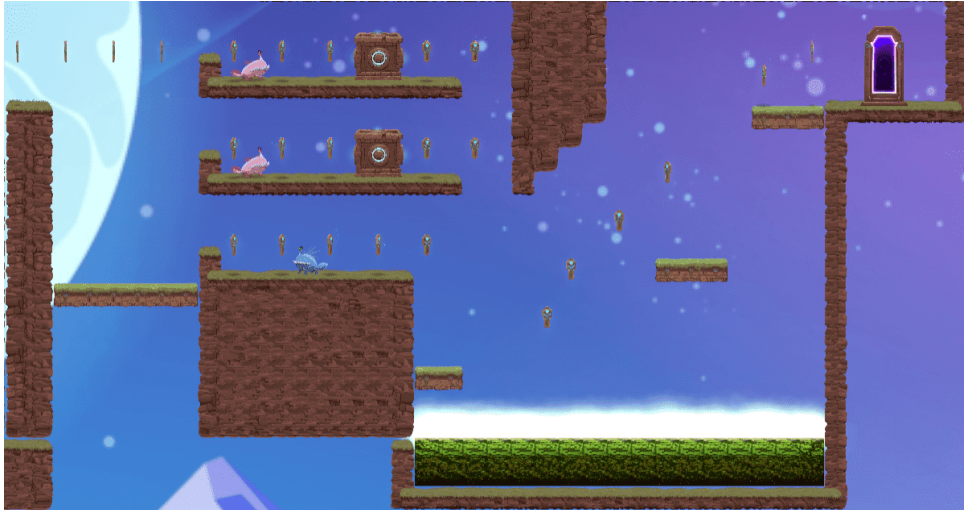


Figura 6.1: Captura de tela de trecho do nível 4 da primeira bateria de testes, demonstrando dificuldade gerada a partir do *design* do nível.

Para criação de níveis para teste de ADD, foi definido que o enfoque estava no movimento, dentre os 3 elementos principais do gênero plataforma [12], em vez do confronto e da interação com o ambiente. Logo, priorizava-se componentes de plataforma ao criar níveis, em vista de sua natureza mais determinista [8]. Além disso, foram feitas modificações da estrutura do jogo, como descrito na Seção 5.4. Também foi definido que cada nível deve ter um único personagem jogável e ao menos um portal de saída e que pode ser composto por segmentos, delimitados por *checkpoints* de modo a contribuir para performance computacional e progresso do jogador.

A seleção de componentes foi feita para evitar aspectos particulares que enviessem as experiências de jogadores diferentes, como a existência de chefes e caminhos paralelos ou opcionais para exploração. Ademais, os níveis possuíam picos de dificuldade destoantes de modo a forçar que as adaptações independente da progressão dos níveis e foram baseados nos níveis dos jogos clássicos do *Super Mario Bros.*

As Figuras 6.3 e 6.4 apresentam um exemplo de nível para teste de ADD, sendo o nível 8 da terceira bateria de testes, respectivamente com a dificuldade originalmente planejada e com a dificuldade adaptada para facilitar o nível a uma dificuldade global



Figura 6.2: Captura de tela de trecho do nível 7 da primeira bateria de testes, demonstrando nível com dificuldade gerada a partir de componentes de plataforma com alta dificuldade individual e baixa densidade.

de 0,2. Ressalta-se que a dificuldade global indica o quanto o nível deve ser modificado para se adequar ao usuário. Nesse contexto, o nível deve ficar mais fácil quanto menor for a dificuldade global, indicando uma menor habilidade do jogador para executar as atividades.

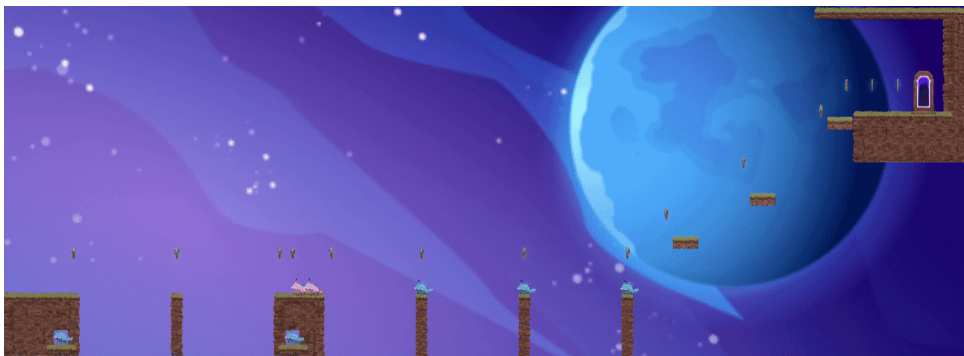


Figura 6.3: Captura de tela de trecho do nível 8 da terceira bateria de testes.

Todos os testes possuíam questionários durante a sessão de jogo, descritos no Apêndice A. Foram aplicados no início e fim da sessão de jogo para, respectivamente, capturar dados sociodemográficos e averiguar as hipóteses de cada bateria, com a descrição do método de ADD e do formato das perguntas descrita na Seção 5.2.

As condições do teste foram especificadas de antemão para assegurar que os participantes tivessem as mesmas informações e chances de completar o jogo, mesmo em ambiente

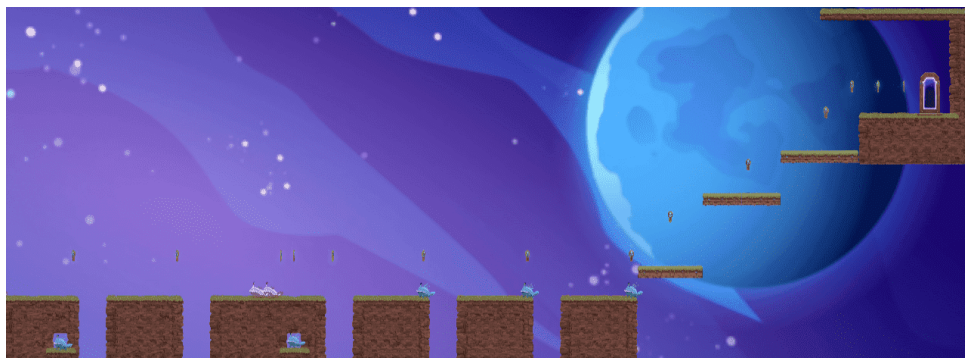


Figura 6.4: Captura de tela de trecho do nível 8 da terceira bateria de testes com adaptação para facilitar o nível ao aplicar dificuldade global de 0,2.

remoto (seja em computador ou celular). Assim, na interface do jogo foram apresentadas as seguintes informações:

1. Apresentação do Termo de Consentimento Livre e Esclarecido (TCLE).
2. Explicação breve da proposta do experimento.
 - Experimento apresentado como um modelo multimétodo de análise de imersão para investigação do impacto de diferentes técnicas de adaptação de jogos, sem informar dados sobre configurações de ajustes que poderiam ser aplicados no jogo.
 - Indicação da quantidade de níveis, do tempo médio de participação e de que é uma pesquisa sem perigos associados.
 - Resumo breve das regras de um jogo do gênero plataforma, apontando que o jogador terá vidas infinitas e a possibilidade de desistir de qualquer nível.
 - Orientação de como deve ser a participação do jogador, pedindo para procurarem um lugar sem muita distração e jogar sem interrupções ou contato externo.
 - Solicitação para as perguntas serem respondidas de forma fidedigna, considerando que o desempenho do participante não está sendo avaliado e que o jogo é parte de um projeto acadêmico, sem intenção comercial.

O volume podia ser ajustado quando desejado, assim como as entradas de controle do avatar entre os modelos disponíveis para cada plataforma. Além disso, antes de iniciar os níveis específicos da bateria, o jogo apresentava o nível de tutorial para contextualizar o jogador na estrutura do jogo.

6.2 Averiguação da estimação de dificuldade e dos dados de jogadores

Na primeira bateria de testes, ao final de cada nível, o jogador foi solicitado a descrever a dificuldade experienciada no nível numericamente em valores de 0 a 10, como ilustrado na Figura 6.5. Assim, constata-se que o valor de dificuldade estimado e o obtido pela média dos jogadores mantêm uma proporção significativa, possuindo resultados superiores ao apresentado com o método anterior de estimativa.

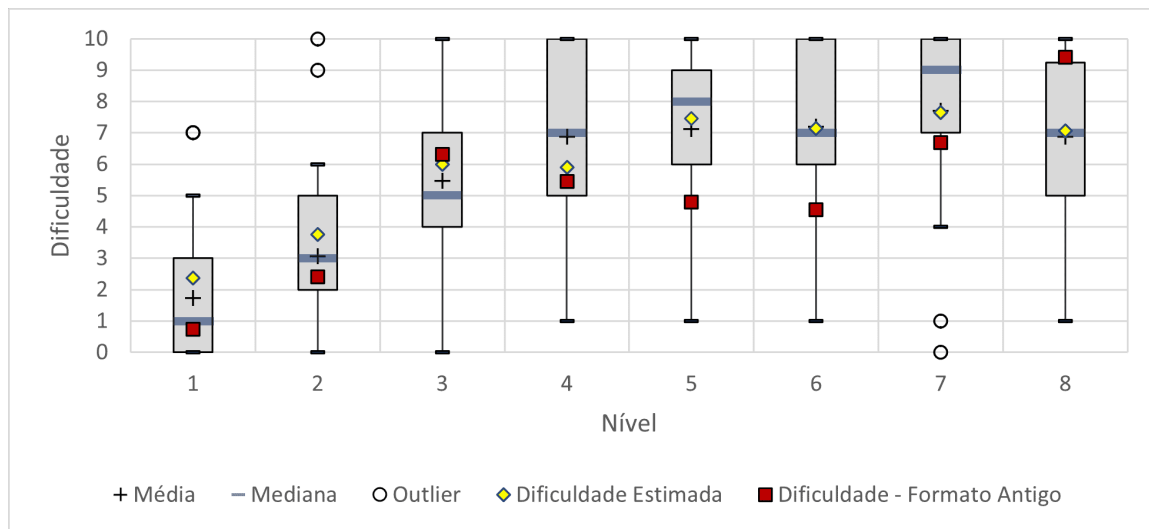


Figura 6.5: Diagrama de caixa com as médias das dificuldades experienciadas pelos jogadores, medidas por meio do questionário, e os valores estimados por algoritmos na primeira bateria de testes. A estimação de dificuldade pelo método atualizado está em amarelo e a estimação pelo método anterior está normalizada e em vermelho.

O valor calculado pelo formato de estimação antigo não possui limites numéricos, como descrito na Subseção 4.2.1, sendo necessário normalizar os valores para comparação. Por outro lado, o medido pelo questionário e estimado pelo método atualizado são limitados entre 0 e 10. Nesse contexto, nota-se que os valores estimados sempre são iguais ou ligeiramente maiores do que a média das dificuldades medidas e estão sempre dentro do primeiro e terceiro quartis, o que não ocorre com o método anterior caso utilizasse componentes que o foco não fosse a movimentação do avatar.

A Tabela 6.1 apresenta os valores de dificuldade estimados e a média dos valores medidos em cada nível, além da porcentagem de participantes que completaram o nível e a probabilidade de significância da diferença entre as dificuldades. Foi feito o teste t de uma amostra para examinar se a média da população é estatisticamente diferente do valor estimado. Nesse cenário, a hipótese nula (de que as dificuldades são iguais) é aceita quão

maior for o valor-p. Ou seja, 5 dos 8 níveis não possuem diferença significativa (hipótese nula aceita e médias iguais) para um nível de significância de 0,05.

Nível	Completado	Dificuldade Estimada	Dificuldade Medida	Valor-p
1	98%	2,36	1,74	0,0441
2	98%	3,76	3,06	0,0593
3	94%	5,98	5,46	0,2101
4	77%	5,90	6,87	0,0345
5	83%	7,46	7,12	0,3803
6	75%	7,13	7,18	0,9002
7	55%	7,64	7,70	0,8851
8	69%	7,06	6,87	0,6718

Tabela 6.1: Porcentagem de participantes que completaram cada nível com as correspondentes dificuldades estimadas, a média das dificuldades medidas e o valor-p da diferença entre os valores de dificuldade na primeira bateria de testes.

Observa-se que a porcentagem de participantes que completam um nível diminui em casos de dificuldade elevada ou com a presença de quebra-cabeças, sendo que o Ajuste Dinâmico de Dificuldade objetiva diminuir a quantidade de desistências. Por sua vez, os valores de dificuldade medidos podem ser influenciados pelo perfil de jogador predominante dos participantes, como visto na Figura 6.6. Ainda que haja uma menor divergência entre participantes do perfil Revolucionário, eles identificam os níveis como mais fáceis do que outros perfis, fazendo as médias diminuírem para cada nível. No caso apontado, as médias e medianas não tiveram impacto nos valores finais, pela baixa porcentagem de participantes desse perfil. Todavia, nota-se a importância em averiguar esse perfil por tenderem a alterar o escopo geral dos resultados.

Outras métricas foram medidas ao final de cada nível, seguindo a escala padrão de valores entre 1 e 7. Assim, foi medido o desafio, a diversão, a frustração e o tédio proporcionados por cada nível, como visto na Figura 6.7. Constata-se que níveis muito difíceis, como o 7, proporcionam maior desafio do que diversão e uma maior frustração para os jogadores. Ademais, a diversão pode crescer, mesmo havendo maior frustração e desafio, como ocorre ao comparar os níveis 1 e 2 com os níveis 3 a 6. Quando muito fácil ou muito difícil, o tédio e a frustração aumentam para a média dos jogadores, assim precisando manter um padrão de dificuldade de acordo com as habilidades do jogador.

Todavia, esse tipo de pergunta ao final de cada nível pode causar contrariedade na interpretação individual da frustração e do tédio, com as respostas dos participantes variando em uma mesma escala. Ou seja, tendem a aumentar os valores de tédio e frustração em conjunto quando sentem dificuldade, mesmo que possam ter conotações

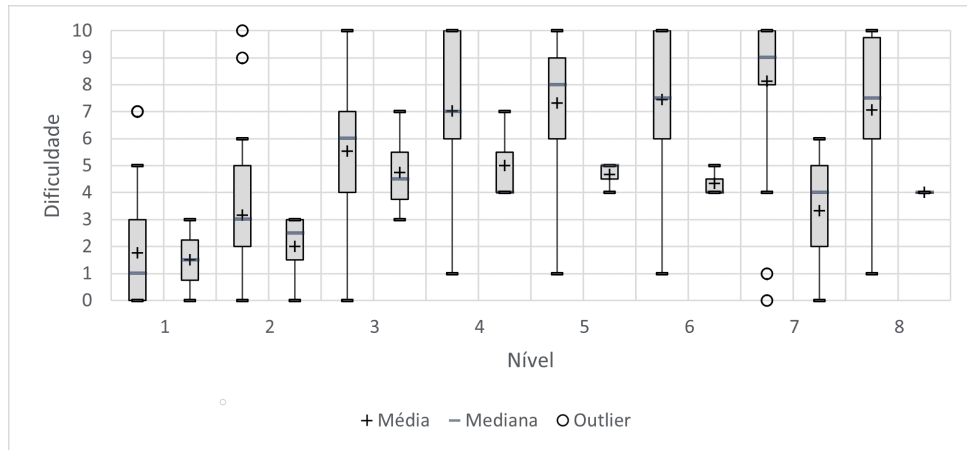


Figura 6.6: Diagrama de caixa com as médias das dificuldades experienciadas pelos jogadores, medidas por meio do questionário e separadas por perfil de jogador predominante (não Revolucionários e Revolucionários) na primeira bateria de testes.

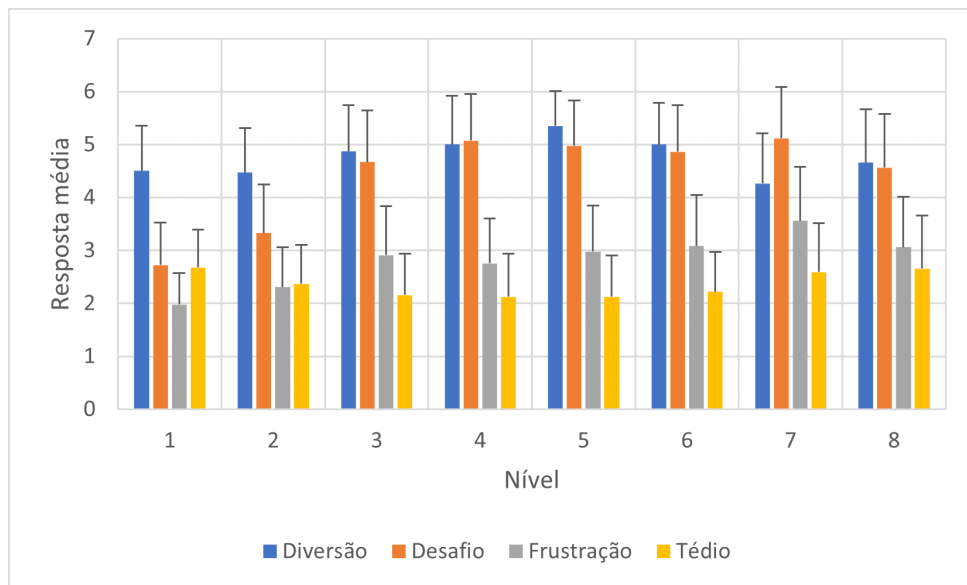


Figura 6.7: Média e desvio padrão das medições de diversão, desafio, frustração e tédio dos participantes na primeira bateria de testes.

contrárias. Isso não se repete quando questionando o jogador apenas ao final do jogo por fazer uma análise mais criteriosa das perguntas nesse momento.

Ainda assim, com o uso do tutorial, nota-se que não houve variação do desafio causado pela apresentação novos componentes aos participantes, como ocorreu nos testes de Rosa [13] e de Santos e Moraes [30]. Conforme analisado em testes anteriores e corroborado pela utilização do tutorial, o primeiro nível deve ser introdutório, com a devida atenção para os distintos componentes serem apresentados e ficar claro para o jogador o que cada um deles representa. Além disso, o tutorial não deve ser adaptado pelo ADD para que o

jogador se familiarize com o jogo, mas dados podem ser coletados para serem utilizados a partir dos nível seguinte.

Ao analisar os níveis individualmente, considerações específicas são identificadas:

1. Básico: teve a dificuldade média quase significativamente igual a dificuldade estimada, com a diferença ocorrendo pela baixa presença de componentes;
2. Nível de plataforma: teve as mesmas considerações do nível básico, porém com menor tédio e maior desafio ao apresentar componentes de plataforma também de dificuldade média;
3. Exploração: utilizou-se da média de dificuldades descrita pela Equação 5.20, com valor entre 5,48 e 6,48. Ademais, apresenta significância entre a dificuldade medida e a estimada a partir do valor médio para o nível e um aumento da diversão e do desafio, ainda que a frustração fosse maior, indicando que ocorrer uma frustração limitada não é necessariamente ruim para o jogador;
4. Quebra-cabeça: teve a maior diferença entre a média e o valor estimado, em vista do método de estimação considerar a dificuldade puramente pelos componentes e não pelo desafio associado ao nível. Ou seja, essa diferença ocorreu pela estimação da dificuldade não considerar desafio proporcionados pelo *design* do nível, sendo uma possível melhora do método para o futuro. Nota-se também uma menor proporção de participantes que completaram o nível e uma maior variação na resposta dos usuários por não ser uma dificuldade de mensuração tão direta, dependendo da compreensão do desafio proposto pelo usuário;
5. Ação: demonstra a importância da inclusão de outros parâmetros para estimação da dificuldade, como inimigos, permitindo abranger o método para outros gêneros de plataforma além do clássico. Observa-se que o método de estimação de dificuldade de inimigos pode ser revisto de acordo com os modelos de inimigos propostos e suas características;
6. Nível de análise de perfil de risco do jogador: corrobora a dificuldade temporal descrita pela Equação 5.13;
7. Baixa densidade de componentes de alta dificuldade: demonstra que níveis de desafio elevado podem diminuir a diversão do usuário, além de impactar na quantidade de desistências, indicando a importância da aplicação de ADD para esses casos;
8. Alta densidade de componentes de média dificuldade: confirma que a alta quantidade de componentes não está diretamente associada à dificuldade, mesmo que

a impacte. Ou seja, a dificuldade média dos componentes do nível tem um maior impacto na dificuldade experienciada pelos jogadores do que o uso em quantidade de componentes, como descrito pela Equação 5.20;

Conforme as análises indicadas acima, avalia-se que a estimação da dificuldade por métodos computacionais consegue prever a dificuldade experienciada pelos jogadores com sucesso, tendo melhorado em relação a resultados anteriores e confirmando a Hipótese 4. Logo, esse método de estimação foi utilizado na aplicação do Ajuste Dinâmico de Dificuldade apresentado na Seção 5.2.4.

Além disso, os participantes também respondiam perguntas ao final da sessão de jogo para analisar as experiências e poder comparar por perfil de jogador. Nesse sentido, o jogador era questionado se considerava que teve um bom desempenho, que havia se arriscado e que teve resiliência ao executar as atividades do jogo, como visto na Figura 6.8. Observa-se que as maiores média da visão do próprio desempenho são para o perfil de Realizador e as menores para Revolucionários, compatível com as respectivas características de buscar maestria e ações disruptivas. Ademais, os Realizadores demonstram se arriscar mais e terem mais persistência, associado a gostarem de desafios e completude de atividades.

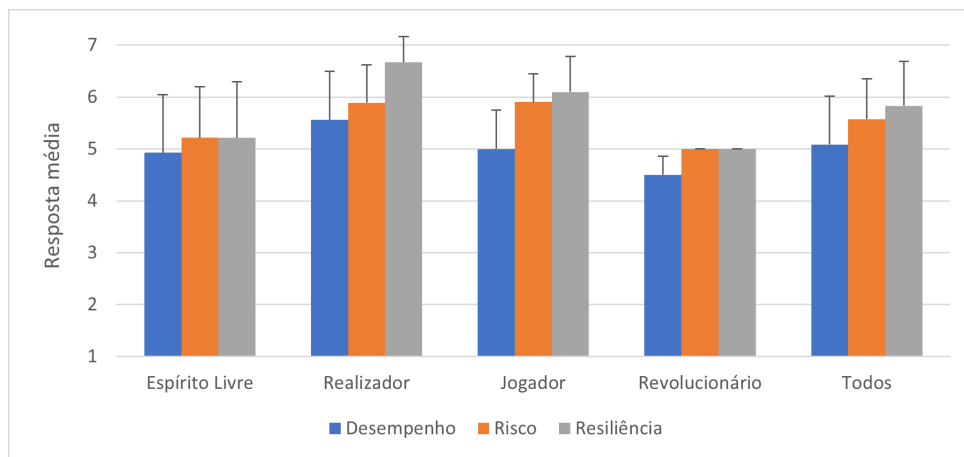


Figura 6.8: Média e desvio padrão das medições de desempenho próprio, perfil de risco e de resiliência dos participantes na primeira bateria de testes, divididas por perfil de jogador.

Distintos componentes do gênero de plataforma impactam a percepção de desafio pelos participantes, como visto na Figura 6.9. Desse modo, identifica-se maior desafio pela distância entre plataforma, corroborando com o fato do ADD ajustar a distância entre plataformas ou a amplitude do salto. Além disso, outras características de plataformas, como largura pequena e movimentação, também impactam no desafio proporcionado,

sendo revisadas suas considerações para o método atualizado de estimação de dificuldade, conforme descrito na Subseção 5.2.2.

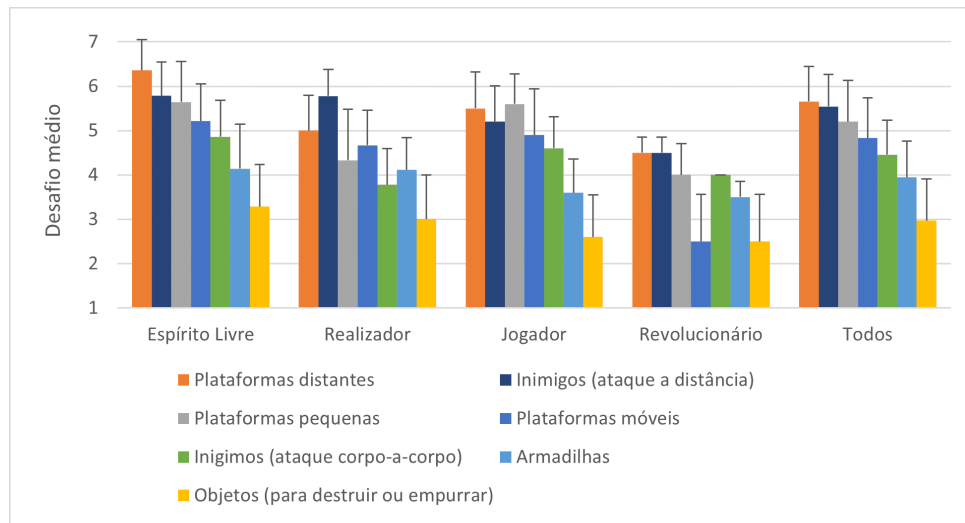


Figura 6.9: Média e desvio padrão do desafio proporcionado por componentes específicos, ordenados de forma decrescente, de acordo com respostas dos participantes na primeira bateria de testes, divididas por perfil de jogador.

Quando analisando perfis de jogadores, os Realizadores são os que têm mais dificuldade com inimigos de ataque a distância. Isso se associa com o fato de buscarem derrotar esses inimigos para maestria do nível, sendo que podem estar em posições de mais difícil acesso do que inimigos de ataque corpo-a-corpo. Ainda assim, os Realizadores, junto aos Revolucionários, mostram uma menor dificuldade em comparação aos outros perfis com os componentes de plataforma por explorarem os níveis e encontrarem outras formas de dificuldade além das triviais. Nesse contexto, componentes de plataforma se associam diretamente a dificuldade padrão de um nível por necessitar dominar e percorrer plataformas para completar os níveis, mas outros componentes podem oferecer maior desafio para participantes que queiram executar atividades complementares dispostas pelo jogo.

Os jogadores também indicaram as ações que influenciaram seu desempenho no jogo, como mostra a Figura 6.10. Assim como em testes anteriores e como esperado pelos objetivos do jogo, o fator de maior influência é o objetivo principal de alcançar o final do nível. Todavia, identifica-se o impacto, ainda que em menor escala, de buscar terminar o nível rapidamente, indicando que dados temporais podem auxiliar na predição da dificuldade experienciada pelo jogador em um nível.

O perfil de Espírito Livre, ao buscar uma diversão mais fácil nos níveis, apresenta o menor impacto dentre os perfis nas atividades de coletar colecionáveis (as chaves no contexto do jogo), derrotar inimigos e explorar o ambiente. Ou seja, não priorizam ações exploratórias no jogo, focando na melhoria do próprio desempenho para concluir os níveis.

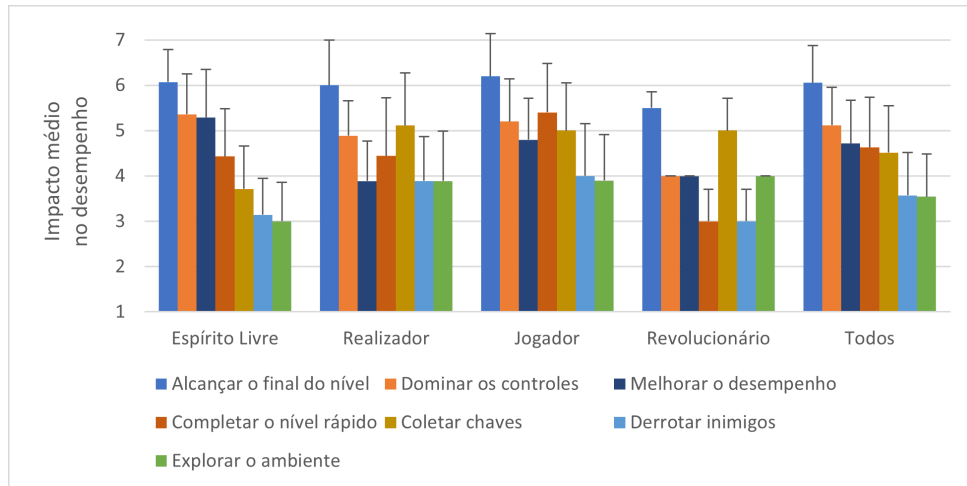


Figura 6.10: Média e desvio padrão das atividade do jogo, ordenadas de forma decrescente, que influenciaram o desempenho dos participantes na primeira bateria de testes, divididas por perfil de jogador.

Por sua vez, o perfil de Jogador, que espera receber recompensas, é o que possui maior impacto na dominação dos controles e em tentar completar os níveis rapidamente. Esses exemplos de características podem ser utilizado para predição dos perfis de jogadores por aprendizado de máquina.

Os colecionáveis e enfrentar inimigos, exceto quando impedem o prosseguimento do jogador, se relacionam à vontade de explorar e a uma gratificação intrínseca ao jogador, especialmente quando não há nenhuma recompensa específica por essas ações. Não foi encontrada uma correlação direta entre a dificuldade do nível e a quantidade proporcional de chaves coletadas ou a quantidade de ataques específicos. Ainda assim, foram as maiores correlações identificadas com os perfis de jogador, como visto nas Figuras 6.11, 6.12 e 6.13.

Para predição do perfil de jogador entre Espírito Livre e Realizador, podem ser usados os parâmetros de proporção de colecionáveis coletados, de ataque a distância utilizados e do tempo explorando andando na direção oposta ao final do nível. No caso, essas ações se relacionam ao perfil de exploração e maestria por indicarem que o jogador está utilizando seu tempo de jogo para executar mais atividades no nível e para fazer ações de modo planejado, o que se observa quando usa ataques a distância em vez de corpo-a-corpo.

Corroborando essas decisões, foi feita uma seleção de recursos supervisionada por filtragem com entrada numérica e saída categórica (Espírito Livre ou Realizador), utilizando a seleção ANOVA. Todos os dados do jogo foram testados, indicados na Seção 4.4.1, e obteve-se testes estatísticos mais relevantes para os parâmetros de porcentagem de chaves coletadas (7,39), tempo andando para direção oposta ao final (3,40) e quantidade de ataques a distância (3,01). Observa-se que usamos proporções ou valores normalizados para os parâmetros da rede neural descritos na Subseção 5.3.1 para minimizar divergências que

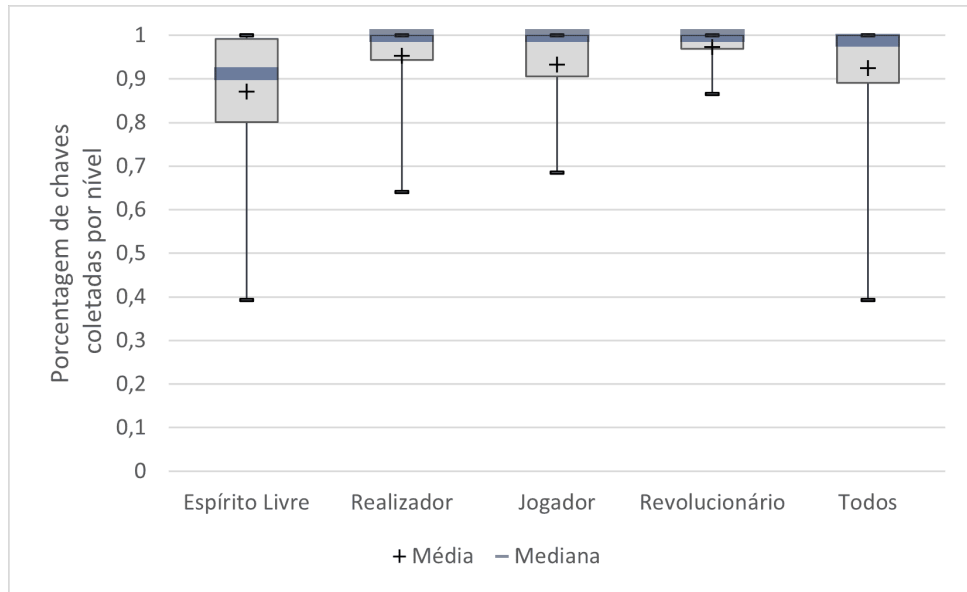


Figura 6.11: Diagrama de caixa com a porcentagem de chaves coletadas, entre 0 e 1, em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.

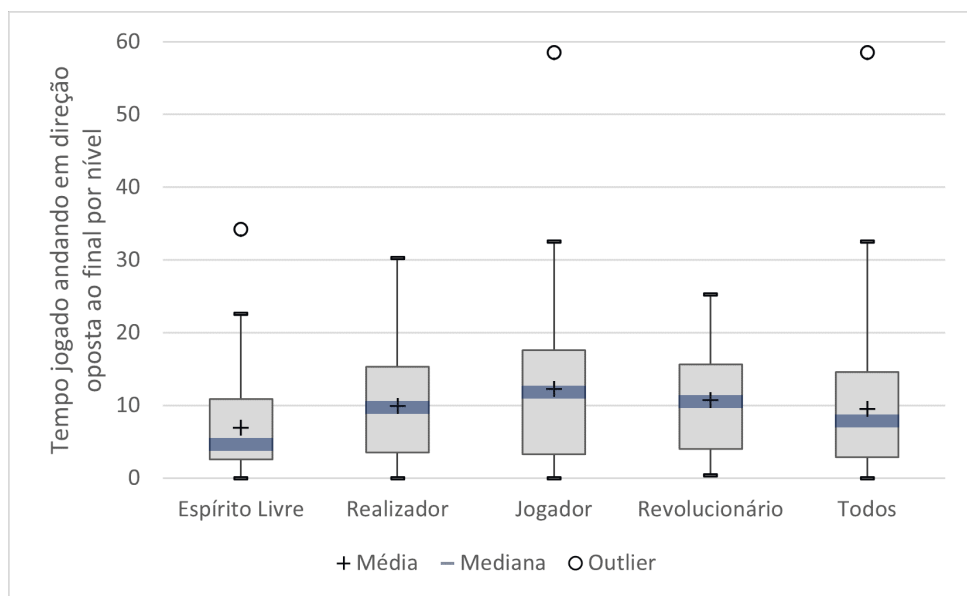


Figura 6.12: Diagrama de caixa com o tempo andando em direção oposta, em segundos, ao final em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.

possam ser dependentes das construções de níveis, pois um nível pode ter mais situações para um certo dado ocorrer do que outros. Logo, é possível identificar variáveis de desempenho que se relacionam significativamente com perfis de jogadores, confirmando a Hipótese 5 e podendo ser utilizada para predição dos perfis por meio de redes neurais.

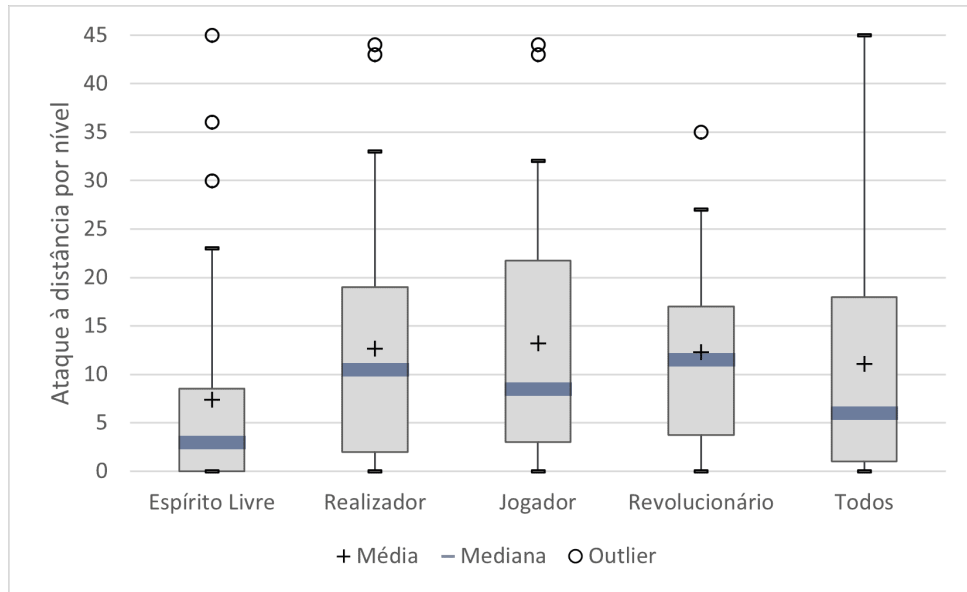


Figura 6.13: Diagrama de caixa com a quantidade de ataques a distância em média por nível pelos participantes na primeira bateria de teste, divididas por perfil de jogador.

De forma complementar, foi analisado os dados que se associavam à dificuldade experienciada pelo jogador, seguindo o padrão utilizado para seleção de parâmetros de perfil de jogador. Assim, foi feita uma seleção de recursos supervisionada por filtragem com entrada numérica e saída numérica (a dificuldade entre 0 e 1), utilizando a seleção Pearson. Todos os dados do jogo foram testados e obteve-se testes estatísticos mais relevantes para os parâmetros de proporção de saltos de sucesso (187,50), tempo jogado (48,57) e quantidade de morte (28,54). As Figuras 6.14, 6.15 e 6.16 apresentam esses dados, respectivamente.

Esses dados foram utilizados para as heurísticas e a rede neural descritas, respectivamente, nas Subseções 5.2.4 e 5.3.1. Constata-se que há uma correlação vista nos gráficos entre a variação dos parâmetros e a dificuldade estimada para cada nível. Isso confirma a Hipótese 6 e indica que os parâmetros podem prever a dificuldade experienciada pelos jogadores ao ser comparada pelo Ajuste Dinâmico de Dificuldade com a dificuldade estimada para o nível a partir dos componentes.

6.3 Comparação entre modelos do Sistema de Controle do ADD

Na segunda bateria de testes, assim como em testes anteriores, corroborou-se a adequação da dificuldade em relação a diminuição da dispersão do desempenho de distintos jogadores. Nesse sentido, foi averiguada a variação de todos os dados coletados do jogo, apresentados

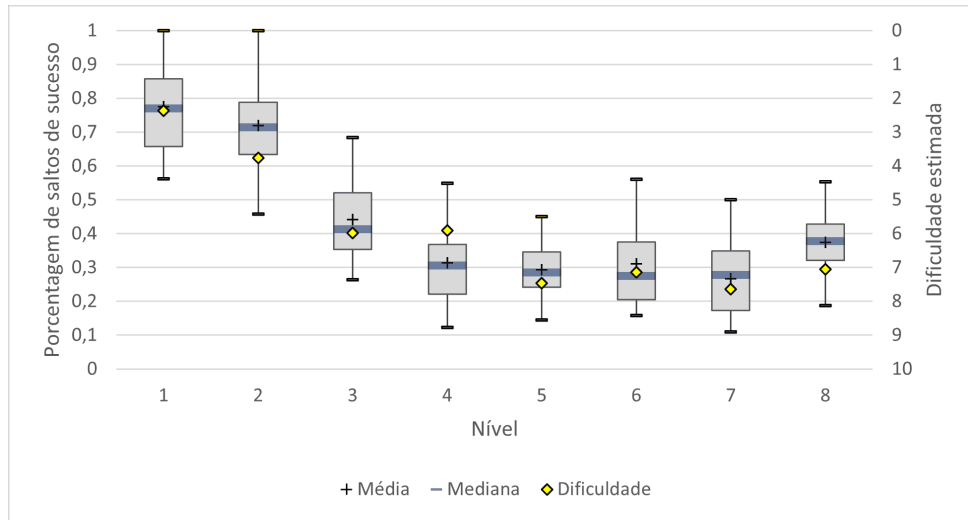


Figura 6.14: Diagrama de caixa com a porcentagem de saltos de sucesso, entre 0 e 1, em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.

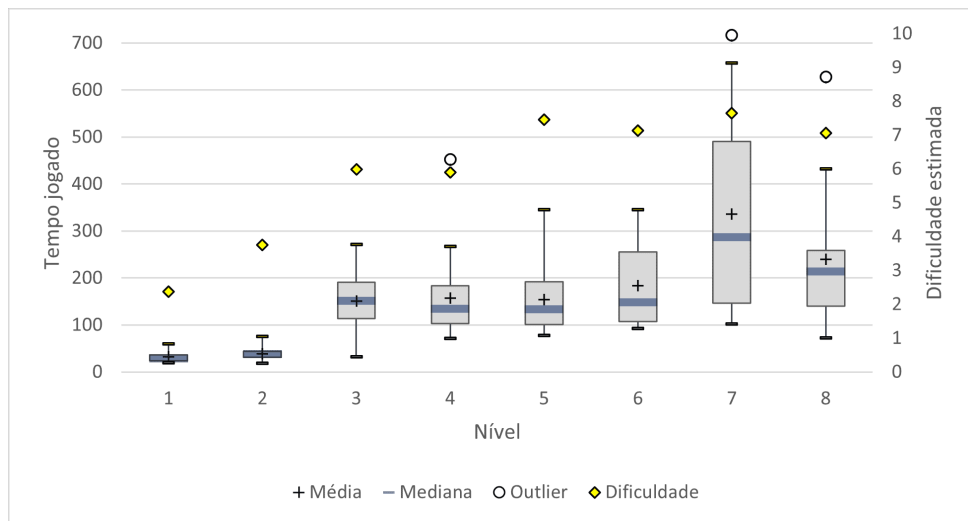


Figura 6.15: Diagrama de caixa com o tempo jogado, em segundos, em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.

na Seção 4.4.1, observando que as médias melhoram e possuem menor desvio quando algum modelo de ADD é utilizado. As Figuras 6.17, 6.18 e 6.19 mostram, respectivamente, os resultados para proporção de saltos de sucesso, quantidade de mortes e tempo jogado em média nos níveis para cada modelo.

Para comparar as médias entre os 4 grupos, foi utilizada a técnica ANOVA, com nível de significância de 0,05. Assim, todos os valores obtiveram variação significativa entre as médias, sendo a hipótese nula (de que as dificuldade são iguais) rejeitada quanto menor for o valor-p. O valor-p para comparação de médias é de 0,04364 para proporção de saltos de sucesso, de 0,00014 para quantidade de mortes e de 0,00012 para o tempo jogado em

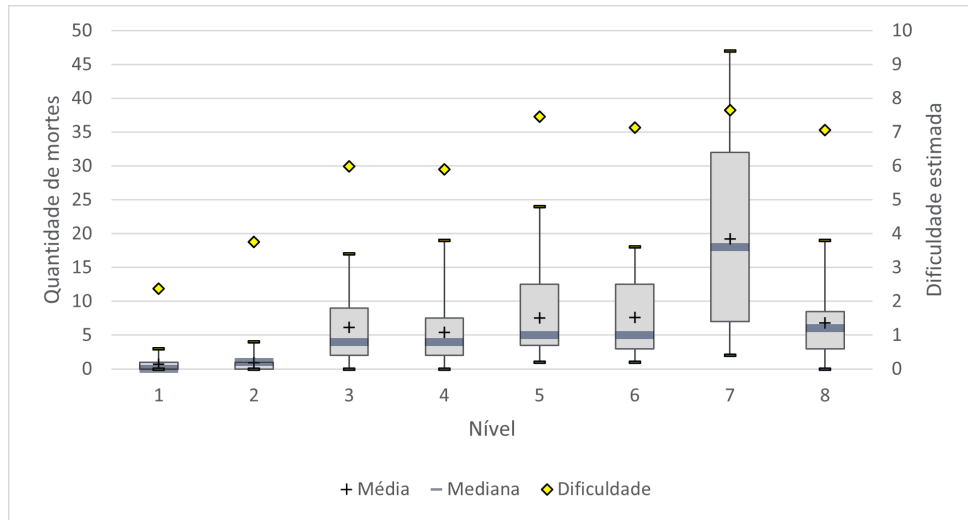


Figura 6.16: Diagrama de caixa com a quantidade de mortes em cada nível pelos participantes na primeira bateria de teste, em comparação a dificuldade estimada.

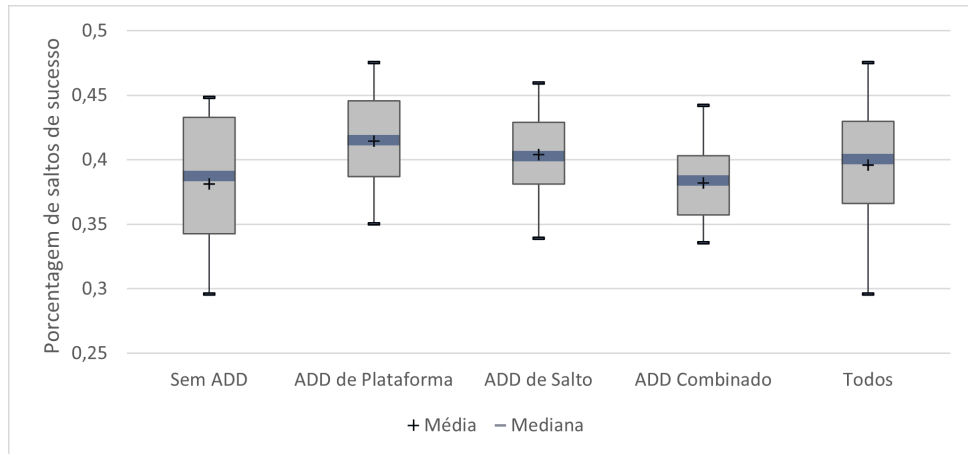


Figura 6.17: Diagrama de caixa com a porcentagem de saltos de sucesso, entre 0 e 1, em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.

média para cada nível pelos participantes.

Observa-se que os níveis foram planejados para durar entre 20 e 90 segundos, como pode ser visto pela variação do tempo jogado, independente do modelo. Ademais, nota-se que níveis sem o ADD possuem uma maior dispersão desse padrão por não adequar a dificuldade ao usuário, variando entre 40 e 150 segundos aproximadamente, enquanto os modelos com ADD estão sempre nessa faixa.

Além disso, constata-se também que o ADD combinado foi o que apresentou menor dispersão entre todos os modelos para todos os dados coletados, seguido pelo ADD de plataforma e o ADD de salto. Isso indica que os modelos eram capazes de adequar a experiência dos jogadores de acordo com suas habilidades, especialmente o modelo de

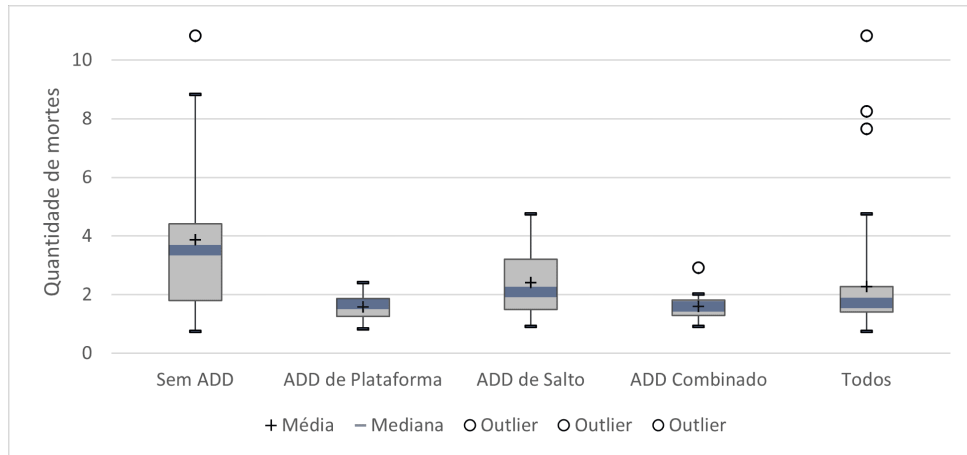


Figura 6.18: Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.

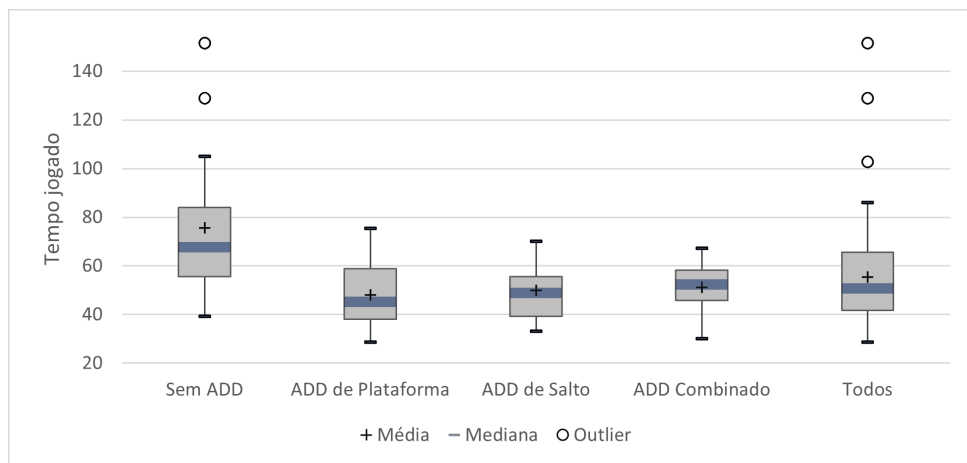


Figura 6.19: Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.

ADD combinado. Inclusive, a médias de morte e tempo jogado foram sempre menores e a média de saltos de sucesso maior quando aplicando algum modelo de ADD.

Em relação a melhoria de desempenho, há mais saltos de sucesso, menos mortes e um menor tempo jogado para o modelo de ADD de plataforma. Ainda que a melhoria do desempenho seja positiva geralmente para os jogadores, especialmente evitando casos de desafio extremos que ocorrem sem ADD, esse não é um indicativo direto de que o modelo seja melhor do que os demais. Um jogo pode somente facilitar para que sempre obtenha melhor resultados dos jogadores. Por sua vez, a diminuição da dispersão entre desempenhos é um fator mais complicado de ser alcançado, pois é diretamente dependente das habilidades dos jogadores.

No caso, jogadores podem preferir desafios mais fáceis (quando de Espírito Livre) ou mais difíceis (quando Realizadores). Com isso, a diminuição da dispersão se torna um melhor indicativo da adequação pela habilidade do que a melhoria dos valores de média. Para melhoria dos métodos, o perfil dos jogadores pode ser utilizado para adequar a dificuldade por esse tipo de característica além do desempenho do jogador.

Ao analisar as mesmas informações apenas para jogadores que preferem jogos com dificuldade fácil, os resultados demonstram apresentar ainda menor divergência para o ADD combinado. Isso pode ser visto nas Figuras 6.20 e 6.21 para quantidade de mortes e tempo jogado, respectivamente. Todavia, a média de mortes é maior ao aplicar o modelo de ADD de salto, podendo indicar um aumento de dificuldade para esse tipo de jogadores ao modificar diretamente uma configuração do avatar. Ademais, observa-se que esse tipo de jogador possui a tendência de se associar mais ao perfil de Espírito Livre, podendo ter resultados superiores ao explorar menos as atividades dos níveis, mesmo que tenham menos habilidade com o jogo.

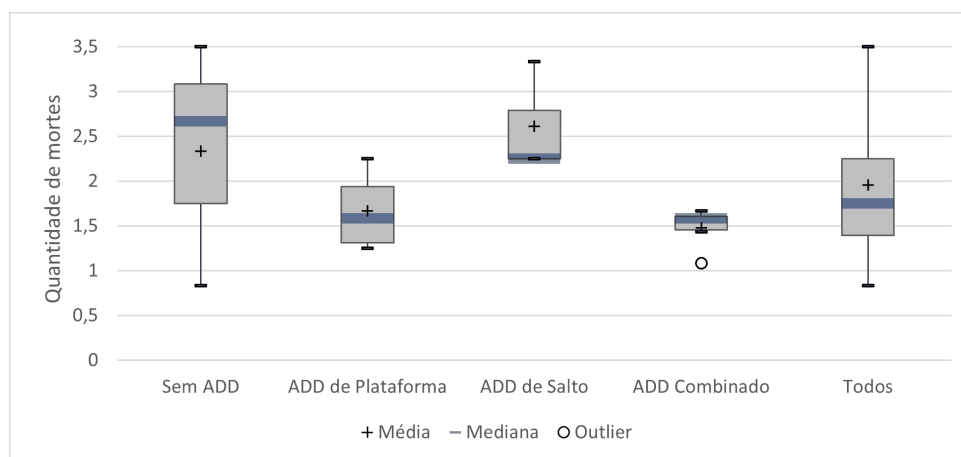


Figura 6.20: Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes com predileção pela dificuldade fácil na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.

De modo complementar, há maior proporção dos níveis completados (terminando sem utilizar a função de desistir do nível) ao utilizar ADD combinado (98%) em comparação ao ADD de plataforma (96%), ADD de salto (90%) e sem ADD (87%). Isso indica que a aplicação de ajustes pelo desempenho pode se associar a resiliência dos jogadores, sendo positivo para o jogo que os usuários não desistam e concluem a experiência por completo.

Além disso, foi feito um acompanhamento das dificuldades por nível para cada modelo, como apresentado nas Figuras 6.22 e 6.23, em ambos os casos variando entre 0 e 1. Respectivamente, analisa-se a progressão da dificuldade final em cada nível para cada modelo que ocorre adaptação de componentes de mundo (ADD de plataforma e combinado) em

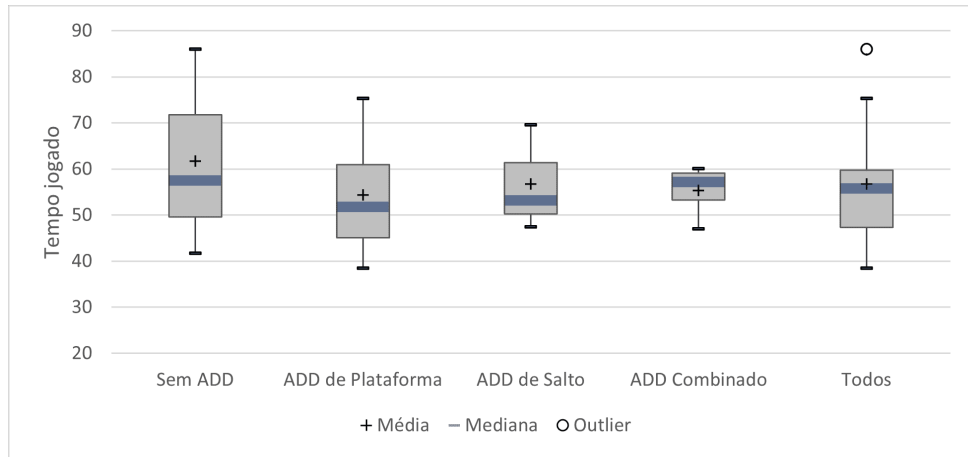


Figura 6.21: Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes com predileção pela dificuldade fácil na segunda bateria de teste, divididas por modelo do Sistema de Controle do ADD.

comparação a dificuldade original de cada nível (sem ADD) e a variação do cálculo da dificuldade global, apresentada na Seção 5.2.4.

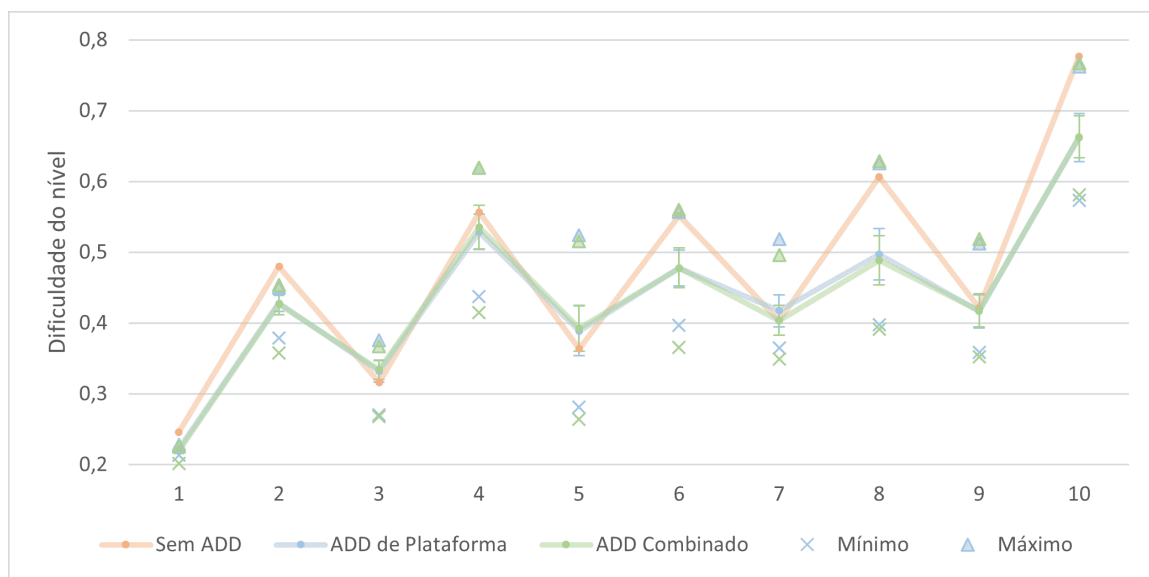


Figura 6.22: Média, desvio padrão, mínimo e máximo da dificuldade de cada nível ao adaptar para os participantes da segunda bateria de testes, divididas por modelo de ADD.

Os modelos de ADD de plataforma e combinado mostraram que as dificuldades tendem a normalizar para os níveis, aproximando de uma progressão e minimizando os extremos de dificuldades propostos pelos níveis dessa bateria. Com isso, os valores de dificuldade global tendem a variar entre 0,3 e 0,6 nesses modelos, com amplas diferenças entre os mínimos e máximos por ser um valor dependente das habilidades de cada usuário. Ainda assim, percebe-se que o jogo com ADD possibilita uma progressão gradativa de dificul-

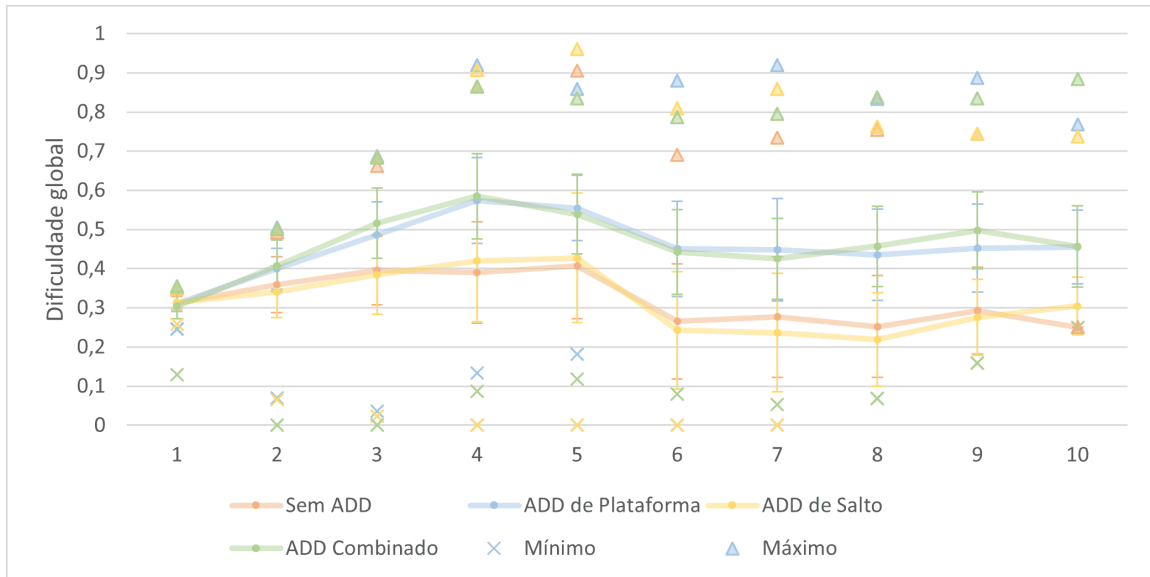


Figura 6.23: Média, desvio padrão, mínimo e máximo da dificuldade global (a qual o nível deveria seguir) para os participantes da segunda bateria de testes, divididas por modelo de ADD.

dade, coerente para os jogadores, sem que o jogador tenham que enfrentar extremos de dificuldade.

Por outro lado, isso não é identificado para os casos de ADD de salto e o jogo sem ADD, indicando que adaptações precisavam ser executadas, geralmente facilitando o jogo para que se adeque ao jogador ao apresentar um baixo desempenho. Com isso em vista, o ADD de salto demonstrou que poderia não ser um ajuste suficiente para impactar na dificuldade experienciada pelos participantes, mas os demais modelos de ADD apresentaram estatísticas positivas para adequação da dificuldade aos jogadores.

As considerações apresentadas acima confirmam a Hipótese 2, com uma melhor adequação às habilidades do jogador quando utilizando métodos de Ajuste Dinâmico de Dificuldade, especialmente com o modelo combinado de adaptação do tamanho de plataformas e da amplitude do salto do avatar.

Ao final da sessão de jogo, foram feitas perguntas sobre a experiência dos jogadores. Nota-se que a maioria das médias foi significativamente diferente ao utilizar a técnica ANOVA com nível de significância de 0,05. Primeiramente, foi averiguado como os jogadores descreviam o próprio desempenho, como ilustrado na Figura 6.24. Nesse cenário, as melhores médias são para o ADD de plataforma (5,52) seguido do ADD combinado (5,42), com um valor-p de 0,06011 e menor variação nas respostas do ADD combinado.

Para participantes com predileção de jogos com dificuldade fácil, como ilustrado na Figura 6.25, a diferença é mais significativa ao comparar o jogo sem ADD (média de 2,66) com os modelos (com média entre 4,25 e 5), havendo menor dispersão para ADD de

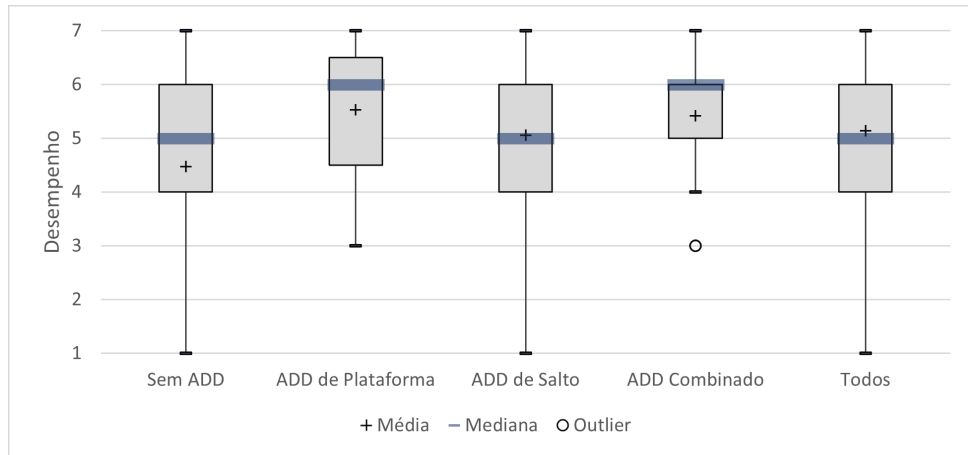


Figura 6.24: Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na segunda bateria de testes, divididas por modelo de ADD.

plataforma e média superior para ADD combinado e de salto. Ainda que não haja uma diferença significativa entre os modelos de ADD, percebe-se uma melhora significativa em relação a quando o ADD não é aplicado na percepção do usuário de ter tido um bom desempenho no jogo, especialmente ao preferir jogos fáceis.

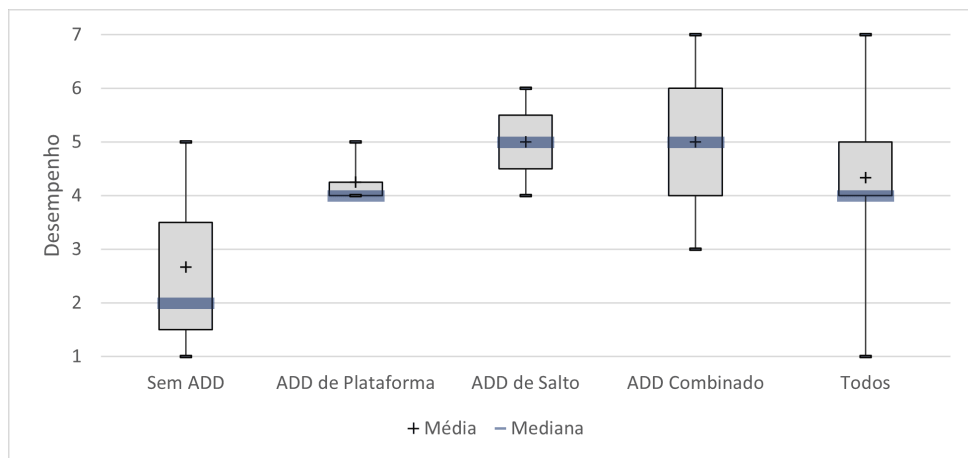


Figura 6.25: Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes com predileção pela dificuldade fácil na segunda bateria de testes, divididas por modelo de ADD.

Em relação aos desafios do jogo, apresentado na Figura 6.26, os valores-p estão abaixo de 0,01, possuindo diferença significativa entre as médias. Nesse contexto, o jogo é considerado mais desafiador com ADD de salto, seguido do jogo sem ADD, com ADD de plataforma e ADD combinado, indicando uma maior dificuldade para domínio dos controles ao adaptar o salto, mas uma melhora dos outros modelos em relação a quando não há ajuste.

Por outro lado, ainda que o jogo fosse desafiador, a maior média da capacidade de superar desafios independente da dificuldade se refere ao ADD de salto, mostrando uma dependência entre esses dois fatores e que o ADD de salto proporciona desafios solucionáveis. O único caso que a média de desafio é maior do que a da capacidade de solucionar se refere ao jogo sem ADD. Ademais, os jogadores mostram gostar mais da experiência de jogar com ADD de plataforma, seguido pelo de salto e combinado, havendo pouca diferença entre essas médias, mas com todas superiores a do jogo sem ADD.

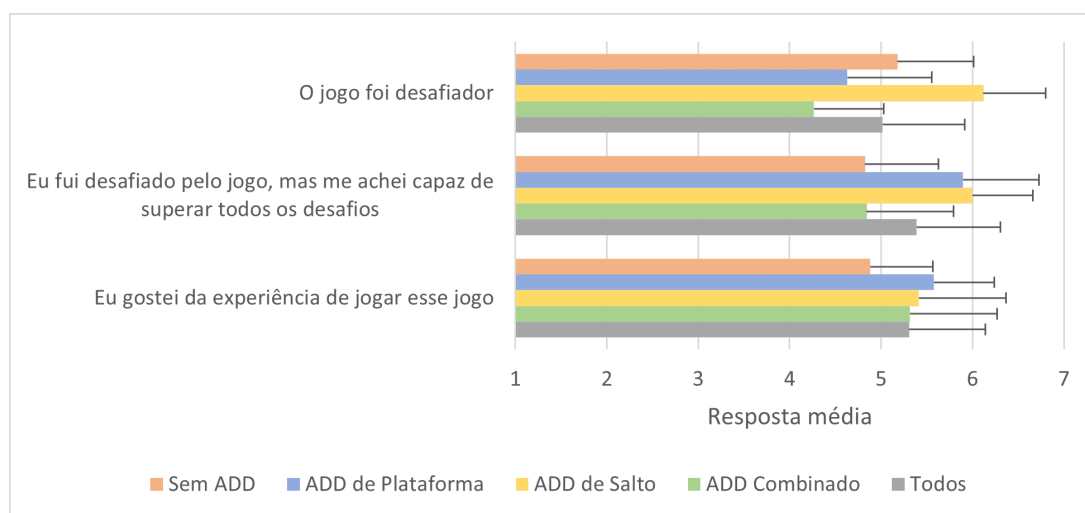


Figura 6.26: Média e desvio padrão das respostas referentes ao desafio proporcionado pelo jogo aos participantes na segunda bateria de testes, divididas por modelo de ADD.

A maior diversão e os menores tédio e frustração ocorrem no jogo com ADD de plataforma, seguido pelo ADD combinado e sem ADD para tédio e frustração e pelo ADD de salto e o ADD combinado para diversão, como visto na Figura 6.27. Considerando que os valores-p apresentam diferença significativa entre as médias ao estarem abaixo de 0,04, nota-se que o ADD de salto pode não ser ideal para utilização em testes futuros ao trazer maior impacto negativo nas respostas dos usuários, enquanto os demais modelos demonstram consistência quando comparados ao jogo sem ADD como parâmetro de base.

Para comparação da dificuldade, foi questionado se o jogo se adequava às habilidades do usuário, se havia muita variação entre níveis seguidos e se a dificuldade aumentava progressivamente, como visto na Figura 6.28. As diferenças são significativas com valores-p abaixo de 0,008 e resultados especialmente piores para o jogo sem ADD, demonstrando melhoria significativa quando os modelos são aplicados na percepção de dificuldade pelos usuários. Como o jogo foi planejado para ter níveis discrepantes, essas perguntas verificam que os modelos de ADD têm sucesso em minimizar essas diferenças extremas e mantêm o jogo com dificuldade progressiva entre os níveis. Ademais, constata-se que os melhores

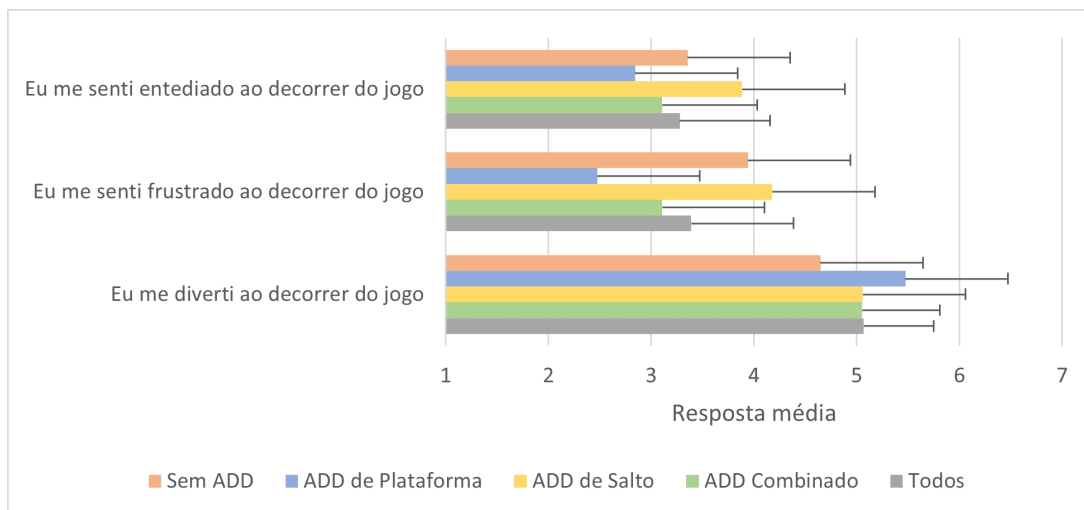


Figura 6.27: Média e desvio padrão das respostas referentes a sentimentos experienciados no jogo pelos participantes na segunda bateria de testes, divididas por modelo de ADD.

resultados de adequação da dificuldade por percepção dos jogadores ocorre para o jogo com ADD de plataforma.

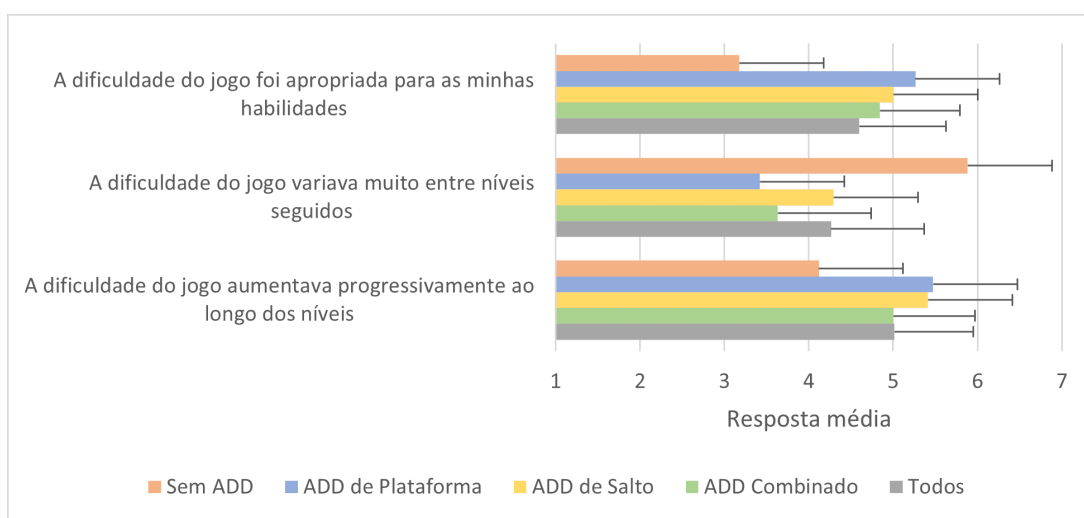


Figura 6.28: Média e desvio padrão das respostas referentes a dificuldade percebida no jogo pelos participantes na segunda bateria de testes, divididas por modelo de ADD.

No escopo de jogadores com predileção por jogos de dificuldade fácil, as diferenças entre as respostas aumenta, como apresentado na Figura 6.29. Nesse sentido, a adequação da dificuldade se mostra ainda mais adequada em comparação ao jogo sem ADD de acordo com as percepções dos usuários, com melhores resultados para o ADD de plataforma.

Ao analisar o estado de fluxo, como visto na Figura 6.30, cinco perguntas foram feitas associando com as principais características de um jogador que mantém nesse estado durante uma sessão de jogo, com significância na diferença entre as médias ao apresentar

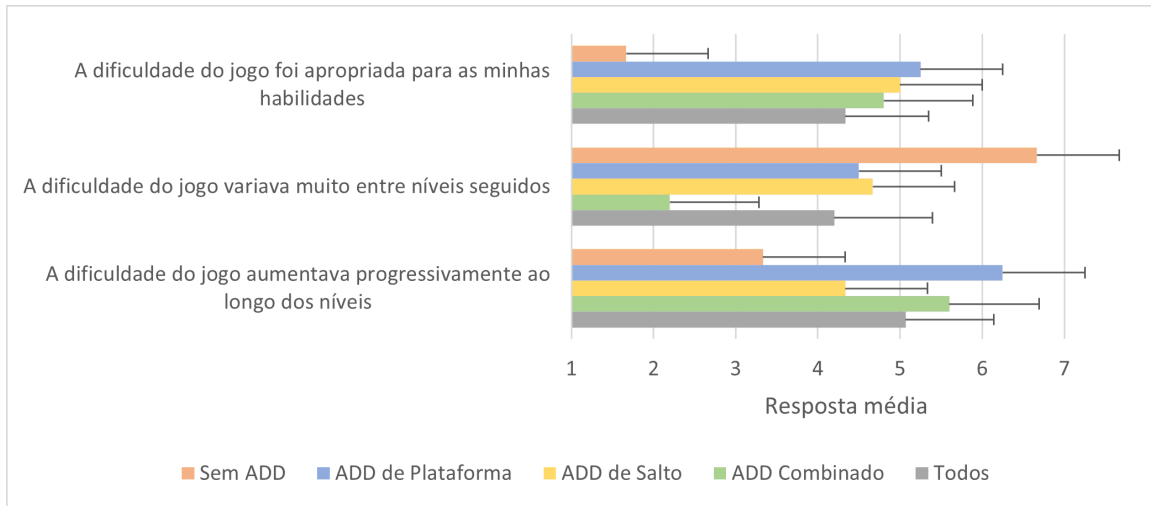


Figura 6.29: Média e desvio padrão das respostas referentes a dificuldade percebida no jogo pelos participantes com predileção pela dificuldade fácil na segunda bateria de testes, divididas por modelo de ADD.

valores-p abaixo de 0,003. Nesse cenário, foi questionado se o jogador se preocupava em falhar, tinha controle de suas ações, perdeu a noção do tempo, teve ações automáticas e estava focado no jogo. As médias sem ADD foram piores em todas as perguntas em comparação aos distintos modelos de ADD, com os melhores resultados variando para cada pergunta entre os modelos.

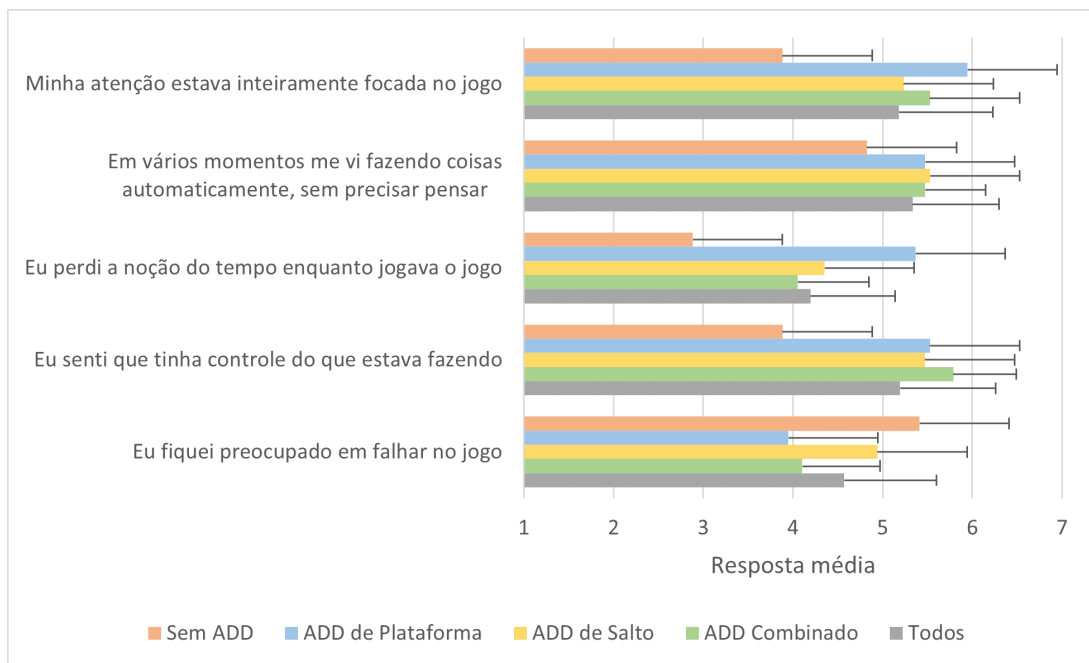


Figura 6.30: Média e desvio padrão das respostas referentes ao estado de fluxo experienciado pelos participantes na segunda bateria de testes, divididas por modelo de ADD.

A Hipótese 3 confirma-se por manter os jogadores em estado de fluxo em um jogo com modelo combinado com adaptação do tamanho de plataforma e da amplitude do salto. Ademais, há significância na diferença entre as médias para o estado de fluxo, o que não havia ocorrido em testes anteriores. Isso também pode ser afirmado para os demais modelos de ADD, com uma melhoria em relação ao jogo sem ajuste e com médias superiores ao valor intermediário, na escala de 1 a 7, para as perguntas referentes ao estado de fluxo. Ou seja, as adaptações não estavam causando variância significativa nos resultados relacionados ao estado de fluxo em testes anteriores pelo estímulo proporcionado pelos níveis não ter uma diferença significativa. Esta alteração obteve sucesso com os novos testes e possibilitou corroborar o formato de medição do estado de fluxo por meio de questionários.

6.4 Comparação entre modelos do Sistema de Análise do ADD

Na terceira bateria de testes, foi verificada a adequação da dificuldade pela diminuição da dispersão do desempenho dos jogadores para os distintos modelos do Sistema de Análise. Assim, a variação de todos os dados coletados do jogo foi analisada a fim de identificar se havia menor desvio para os modelos de ADD. Dentre os dados descritos na Seção 4.4.1, os que tiveram maior diminuição da dispersão e melhoria da média dos participantes foram para quantidade de mortes, quantidade de saltos falhos, quantidade proporcional de saltos de sucesso, quantidade proporcional de chaves coletadas e tempo jogado, respectivamente apresentados nas Figuras 6.31, 6.33, 6.34, 6.35 e 6.19

Para comparar as médias entre os 3 grupos, foi utilizada a técnica ANOVA, com nível de significância de 0,05. Nesse contexto, todos os valores obtiveram variação significativa entre as médias, sendo a hipótese nula (de que as dificuldades são iguais) rejeitada por valores-p pequenos. O valor-p para comparação de médias é de 0,00053 para quantidade de mortes, de 0,02516 para o tempo jogado, de 0,00702 para quantidade de saltos falhos, de 0,02615 para proporção de saltos de sucesso e de 0,21181 para quantidade de chaves coletadas em média para cada nível pelos participantes.

Como visto em testes anteriores, os dados de desempenho apresentaram melhoria de performance dos jogadores e menor dispersão ao aplicar algum modelo de ADD. Especificamente sobre a quantidade de mortes e o tempo jogado, percebe-se uma redução da média para completar cada nível e uma menor dispersão quando aplicado o ADD de RNA. Observa-se que os níveis foram planejados para durar entre 40 e 90 segundos, independente do modelo, o que não ocorre em todas as situações quando não há aplicação de ADD. Isso indica que os modelos eram capazes de adequar a experiência dos jogadores de acordo com suas habilidades, especialmente o modelo de ADD de RNA.

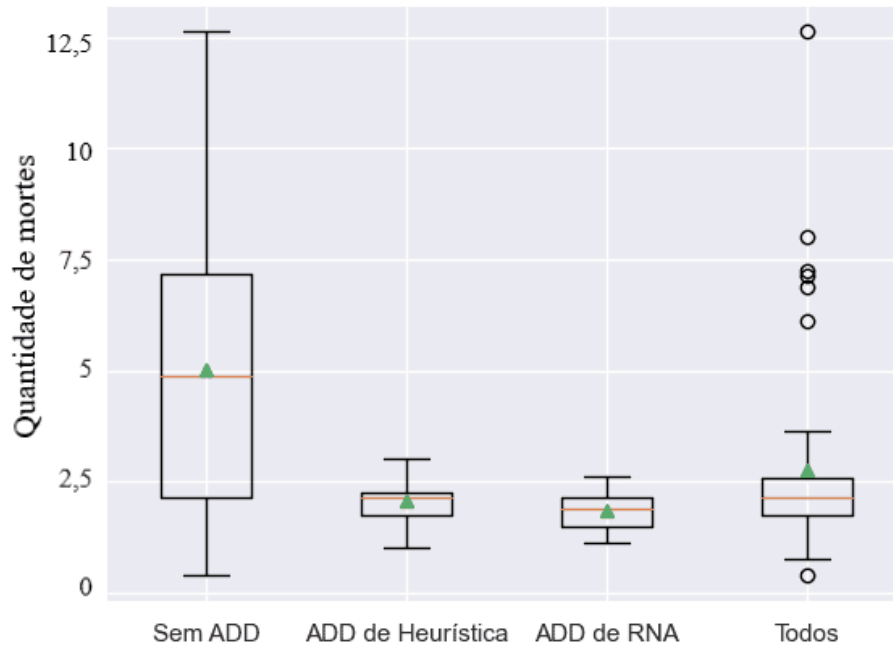


Figura 6.31: Diagrama de caixa com a quantidade de mortes em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.

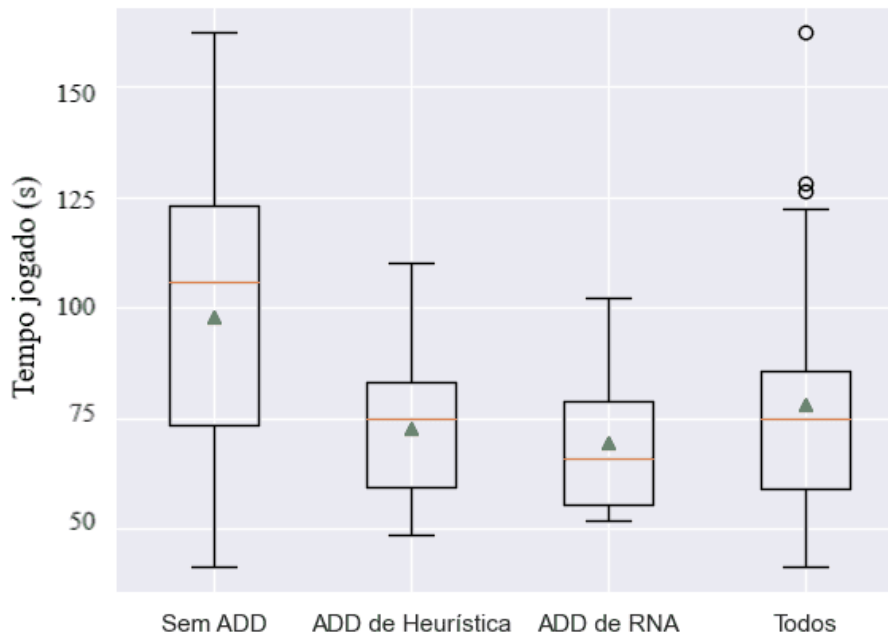


Figura 6.32: Diagrama de caixa com o tempo jogado, em segundos, em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.

Em relação a outros aspectos do desempenho, também foi observado uma redução da quantidade de saltos de sucesso e um aumento da porcentagem de saltos de sucesso em

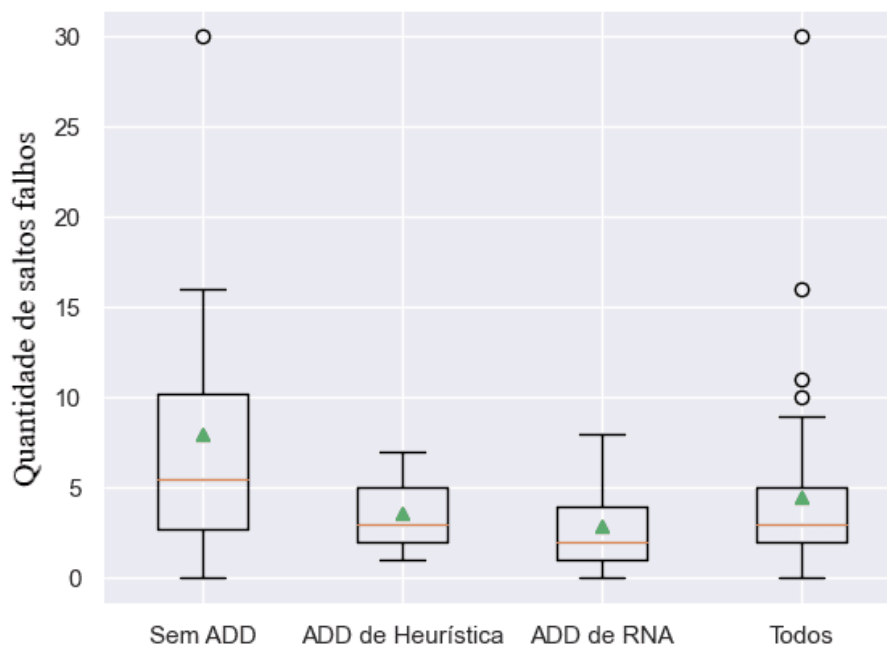


Figura 6.33: Diagrama de caixa com a quantidade de saltos falhos em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.

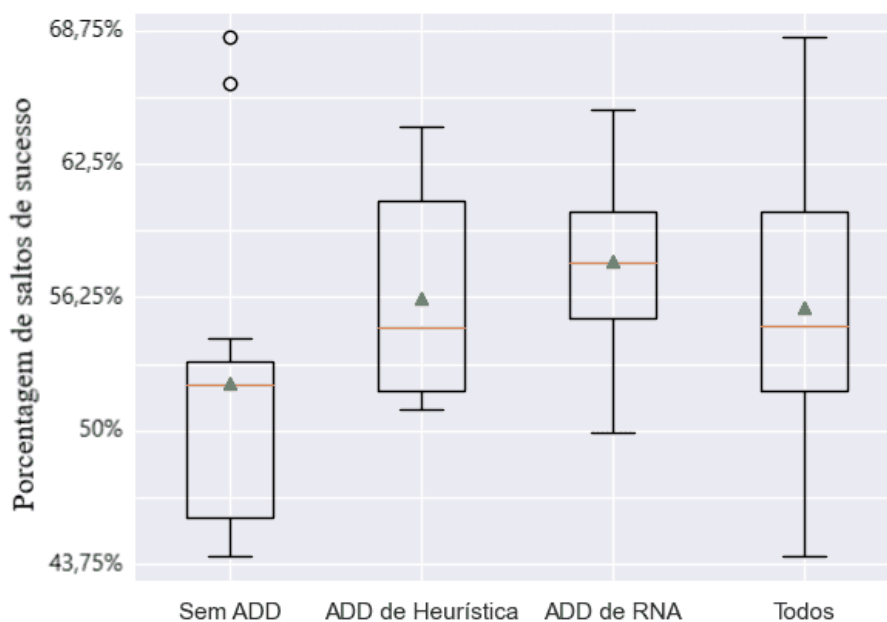


Figura 6.34: Diagrama de caixa com a porcentagem de saltos de sucesso em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.

um nível ao aplicar ADD. Por sua vez, um dado que teve melhoria apenas com o ADD de RNA foi a porcentagem de chaves coletadas, associado a adequação da progressão do desafio pelo perfil do jogador e motivando a completar tarefas adicionais dos níveis.

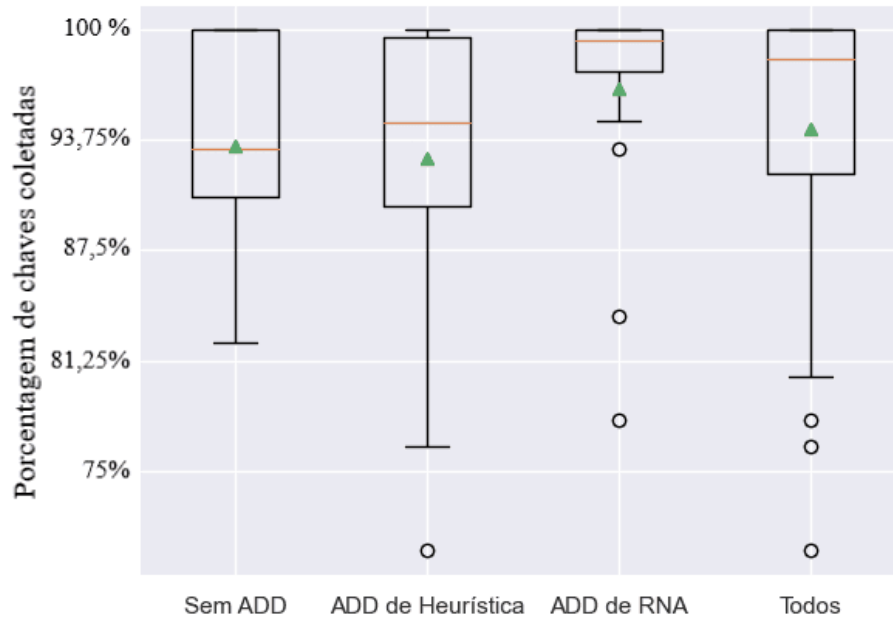


Figura 6.35: Diagrama de caixa com a porcentagem de chaves coletadas em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Análise do ADD.

Ao analisar os dados de acordo com o perfil dos jogadores, é possível observar como são executados desafios propostos pelo jogo, como visto nas Figuras 6.36 e 6.37 para porcentagem de saltos de sucesso e porcentagem de chaves coletadas, respectivamente. Para esse tipo de análise, os demais perfis Hexad foram desconsiderados e os participantes foram divididos conforme a predominância entre Espírito Livre (54% dos participantes) e Realizador (46% dos participantes).

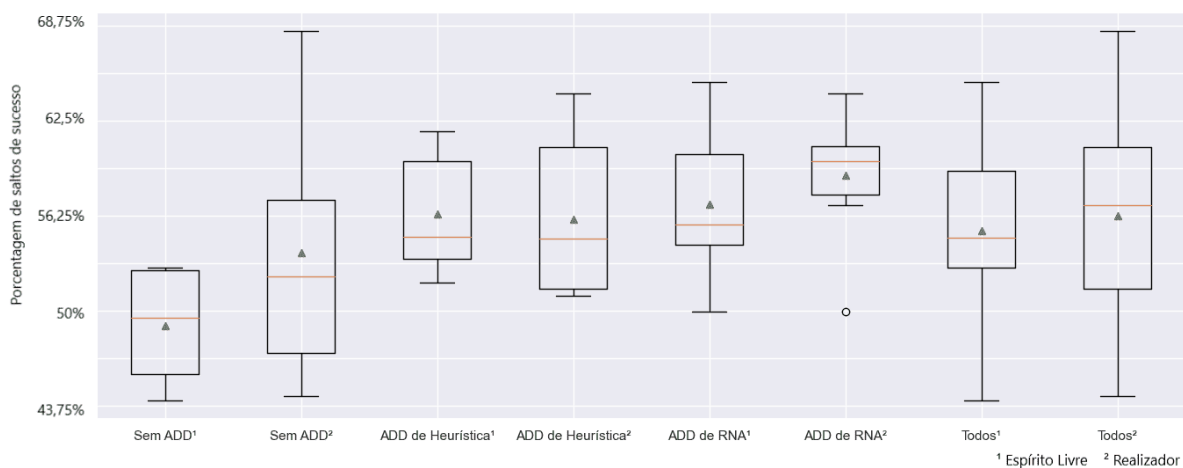


Figura 6.36: Diagrama de caixa com a porcentagem de saltos de sucesso em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Controle do ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.

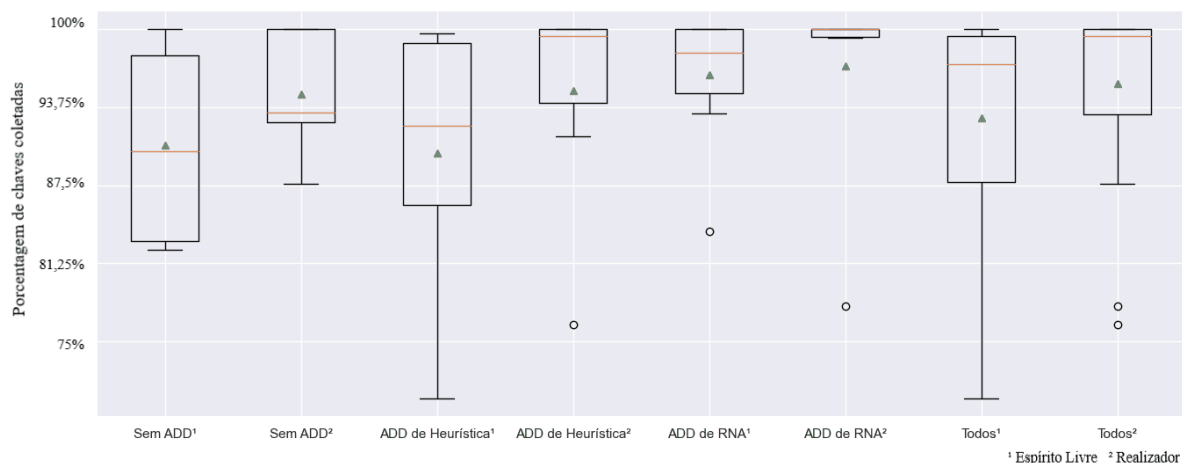


Figura 6.37: Diagrama de caixa com a porcentagem de chaves coletadas em média por nível pelos participantes na terceira bateria de teste, divididas por modelo do Sistema de Controle do ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.

Nota-se que a porcentagem de saltos de sucesso é maior para jogadores do perfil Realizador quando não há aplicação de ADD e que essa média aumenta para ambos os perfis quando há algum modelo aplicado. Inclusive, as médias são aproximadas independente do perfil do jogador, em vista da adequação do jogo à habilidade. Além disso, quando há a aplicação do ADD de RNA há uma maior média e menor dispersão para o perfil Realizador, os quais buscam maestria.

Por outro lado, observa-se que o ADD de heurísticas mantém uma ampla variação na porcentagem de chaves coletadas para o perfil de Espírito Livre, ainda que haja melhora para o perfil Realizador. Todavia, isso é melhorado para ambos os perfis ao aplicar o ADD de RNA. Ademais, houve uma grande melhora na completude das tarefas adicionais do nível com RNA, especialmente considerando que a grande maioria dos participantes do perfil Realizador coletou a maioria das chaves nos níveis e com baixa dispersão dos desempenhos.

De modo complementar, há maior proporção dos níveis completados (terminando sem utilizar a função de desistir do nível) ao utilizar ADD de RNA (98%) em comparação ao ADD de heurísticas (95%) e sem ADD (87%). Isso indica que a aplicação de ajustes pelo desempenho pode se associar a resiliência dos jogadores, sendo positivo para o jogo que os usuários não desistam e concluem a experiência por completo.

Ademais, o acompanhamento das dificuldades foi feito por nível para cada modelo testado. As Figuras 6.38 e 6.39 mostram, respectivamente, a dificuldade média de cada nível e a variação da dificuldade global apresentada na Seção 5.2.4, ambas variando entre 0 e 1. Como visto em testes anteriores, as dificuldades tendem a sair dos extremos propostos originalmente pelos níveis sem ADD, normalizando as dificuldades para uma progressão

mais gradativa e em conformidade com o jogador. Ocorre uma ampla diferença entre os mínimos e máximos de cada nível pela dificuldade variar de acordo com cada jogador.

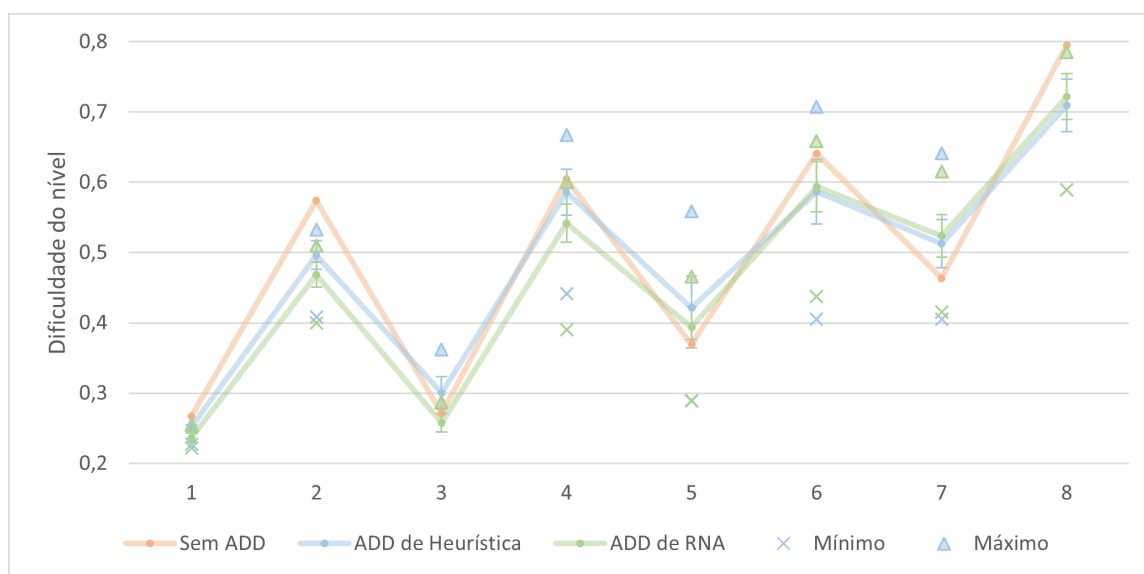


Figura 6.38: Média, desvio padrão, mínimo e máximo da dificuldade de cada nível ao adaptar para os participantes da terceira bateria de testes, divididas por modelo de ADD.

Ao comparar a progressão da dificuldade entre os modelos de heurísticas e de RNA, observa-se uma variação entre mínimos e máximos ligeiramente menor para o caso de RNA. Além disso, com RNA a dificuldade começa com valores inferiores e cresce para valores superiores de dificuldade em comparação a modelo de heurísticas, indicando uma progressão mais gradativa na escala de dificuldade do jogo.

A média da dificuldade global no jogo sem ADD tende a ficar baixa em todos os níveis pelos usuários terem desempenhos inferiores nesse caso. Enquanto isso, ambos os modelos de ADD se aproximam mais de um intermediário de dificuldade para os participantes, sem os extremos vistos sem nenhum ADD, e com o modelo de RNA possuindo uma progressão mais linear.

Ao comparar os dados de dificuldade com o perfil do jogador entre Espírito Livre e Realizador, nota-se que a dispersão da média de dificuldade do jogo é menor em ambos os modelos de ADD em relação aos níveis sem ADD para qualquer perfil. Isso indica que os jogadores poderiam estar com o desafio destoante às habilidades ao não aplicar ajustes. A Figura 6.40 apresenta a média de dificuldade de cada nível para cada perfil de jogador, conforme medido por questionário no início do jogo e dividido somente entre os perfis Espírito Livre e Realizador.

Observa-se que a dificuldade se mantém próxima independente do perfil para o modelo de heurísticas. Para jogadores Espírito Livre no modelo de RNA, a dificuldade tem uma progressão mais gradativa ao começar abaixo e terminar ligeiramente superior a média do

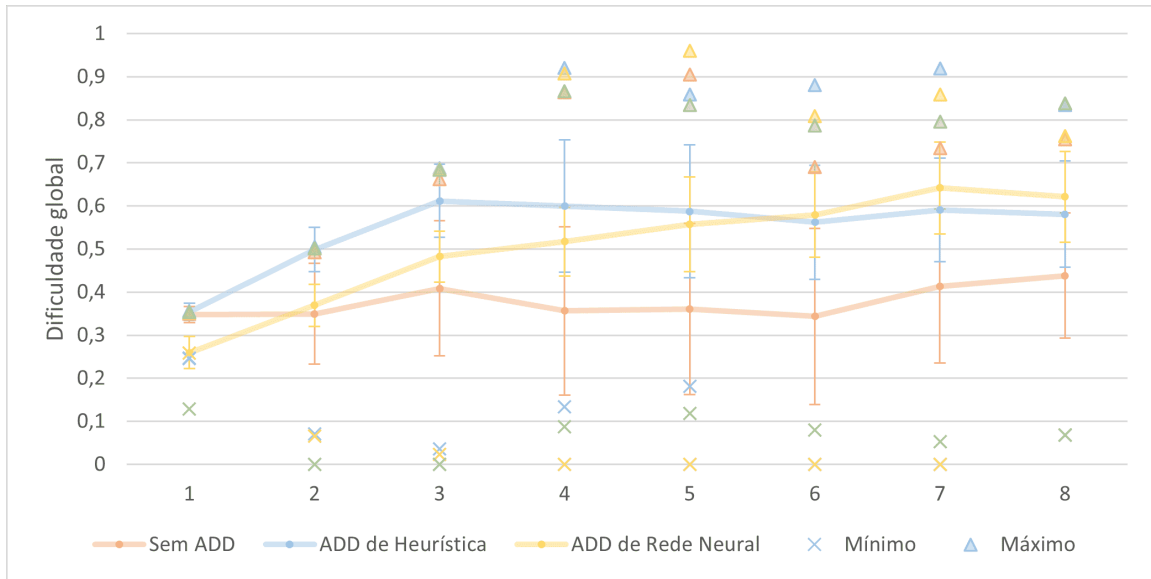


Figura 6.39: Média, desvio padrão, mínimo e máximo da dificuldade global (a qual o nível deveria seguir) para os participantes da terceira bateria de testes, divididas por modelo de ADD.

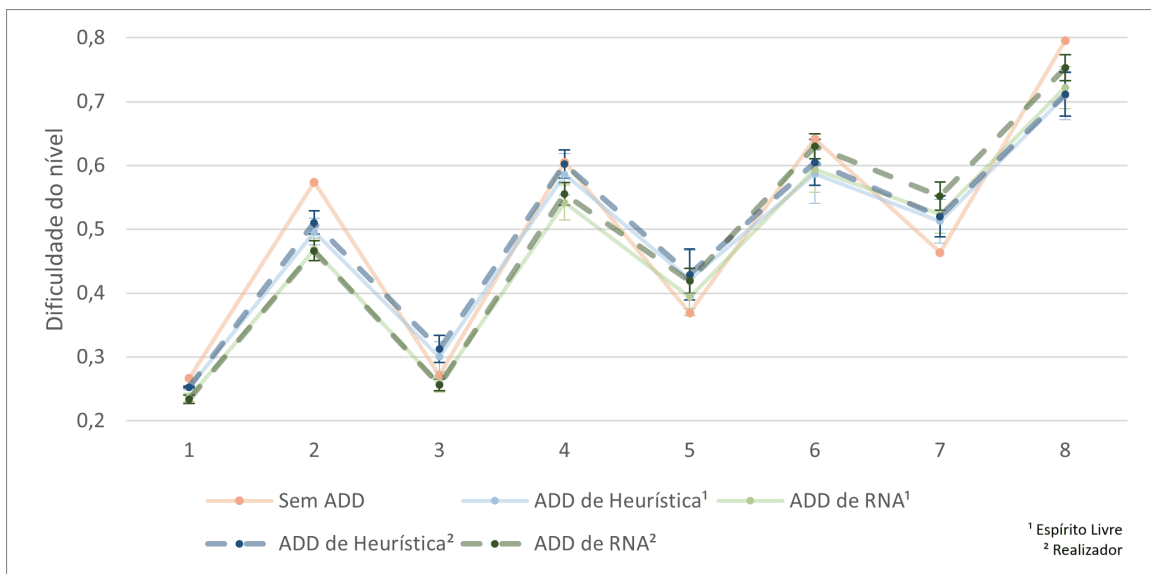


Figura 6.40: Média e desvio padrão de cada nível ao adaptar para os participantes da terceira bateria de testes, divididas por modelo de ADD pela cor e pelo perfil de jogador predominante entre Espírito Livre e Realizador pelo tracejado das linhas.

modelo de heurísticas. Por sua vez, a dificuldade possui uma média superior às demais a partir da metade do jogo para o perfil Realizador no modelo de RNA, indicando que o perfil foi identificado com sucesso e o jogo alterado em correspondência, apresentando desafios mais difíceis.

Um questionário foi aplicado ao final de cada sessão de jogo sobre a experiência dos jogadores. Todas as médias foram significativamente diferentes ao utilizar a técnica ANOVA com nível de significância de 0,05. Em primeira instância, foi averiguado como os jogadores descreviam o próprio desempenho, como apresentado na Figura 6.41. O modelo de RNA teve uma média superior (6,36) ao de heurísticas (5,65) e a sem ADD (4,24), com um valor-p de 0,000011.

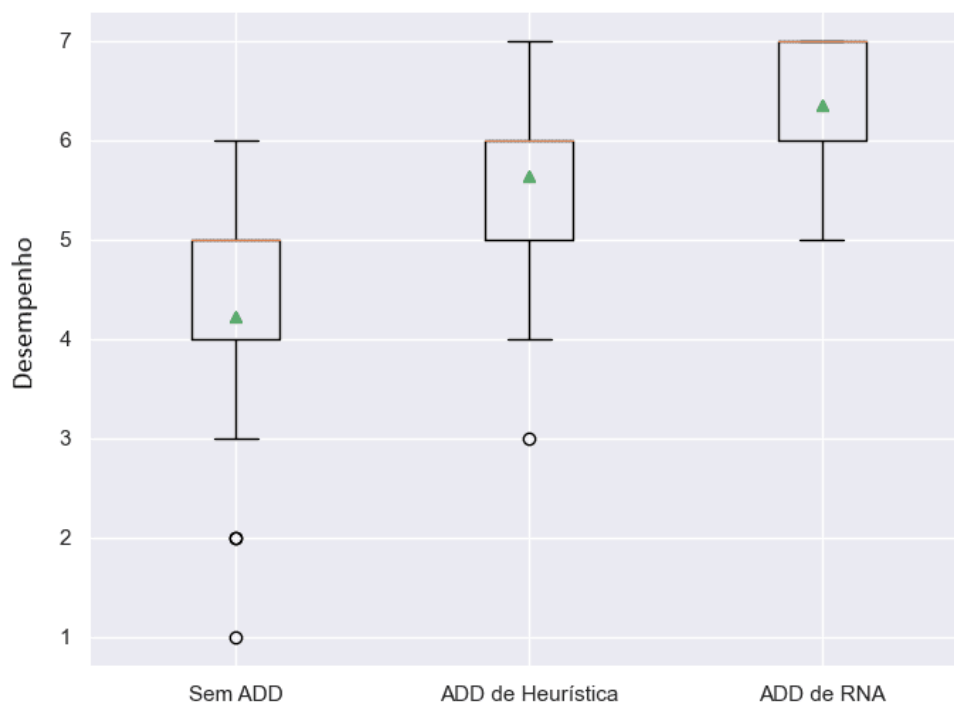


Figura 6.41: Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na terceira bateria de testes, divididas por modelo de ADD.

Além da percepção própria do desempenho ter sido melhor para o ADD de RNA, foi possível identificar que os jogadores do perfil Realizador possuem uma média superior e uma menor dispersão de percepção do desempenho aos de Espírito Livre. Como ilustrado na Figura 6.42, ainda que haja essa diferença entre os perfis, houve uma melhora a percepção de desempenho para os modelos de ADD em ambos os perfis, especialmente com o ADD de RNA.

Em relação a dificuldade do jogo, os participantes foram questionados se estava adequada às habilidades, se variava muito entre níveis seguidos e se aumentava progressivamente no decorrer do jogo, assim como visto na Figura 6.43. Nesse contexto, a dispersão foi menor e a média foi maior para o modelo de RNA referente à adequação (6,53) em comparação ao modelo de heurísticas (5,83) e sem ADD (3,12). Nota-se uma ampla melhora da média, com um valor-p inferior a 0,000001, indicando o impacto positivo que o Ajuste Dinâmico de Dificuldade pode ter na experiência do usuário.

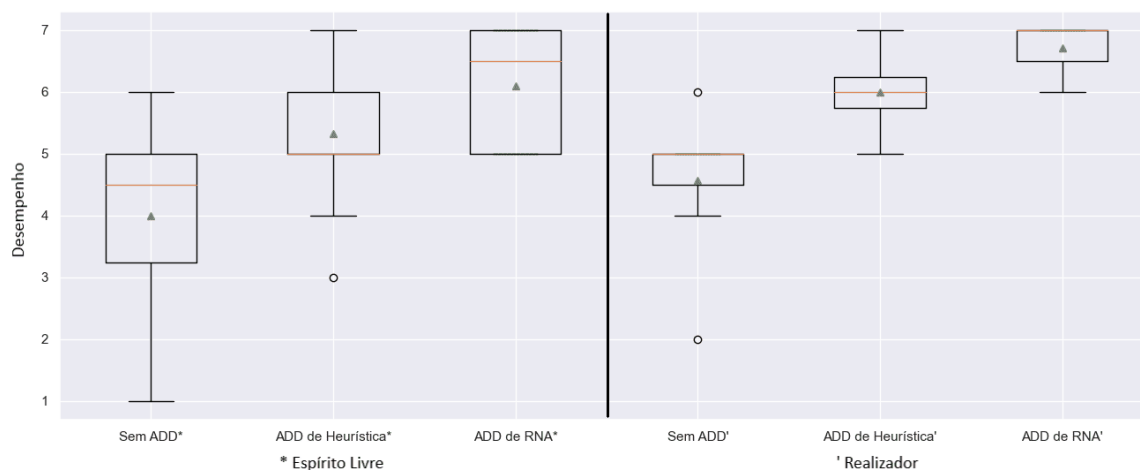


Figura 6.42: Diagrama de caixa das respostas referentes ao desempenho próprio percebido pelos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.

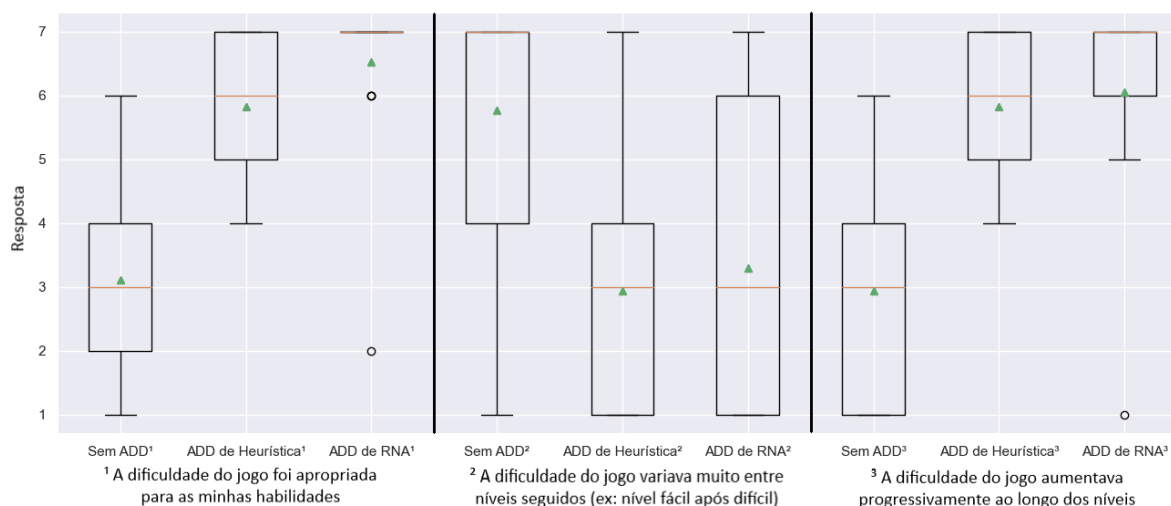


Figura 6.43: Diagrama de caixa das respostas referentes a dificuldade percebida no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD.

A variação de dificuldade entre níveis também foi reduzida ao aplicar ADD, com valor-p de 0,000556. Todavia, houve ainda grande dispersão das diferenças entre percepções entre distintos jogadores, podendo reduzir isso com o aprimoramento do ADD ao utilizar outros métodos de aprendizagem de máquina e incluir outros perfis de jogadores na análise. Em contraponto, a progressão da dificuldade teve melhoria na média para os modelos de RNA e de heurísticas, nessa ordem, incluindo uma menor dispersão das respostas, com valor-p inferior a 0,000001. Ou seja, o ADD foi capaz de impactar a percepção do jogador do jogo possuir uma dificuldade crescente e gradativa, mesmo que os níveis originalmente não fizessem isso.

De forma adicional, a pergunta sobre a dificuldade ser apropriada ao jogador, assim

como a maior parte do restante do questionário, não teve variação significativa ao comparar separadamente jogador com perfil predominante de Espírito Livre ou Realizador. Isso pode ser visto na Figura 6.44. Há uma dispersão ligeiramente menor nas respostas para o perfil Realizador, mas uma média próxima independente do perfil, sendo sempre maior quando há ADD aplicado, especialmente no ADD de RNA.

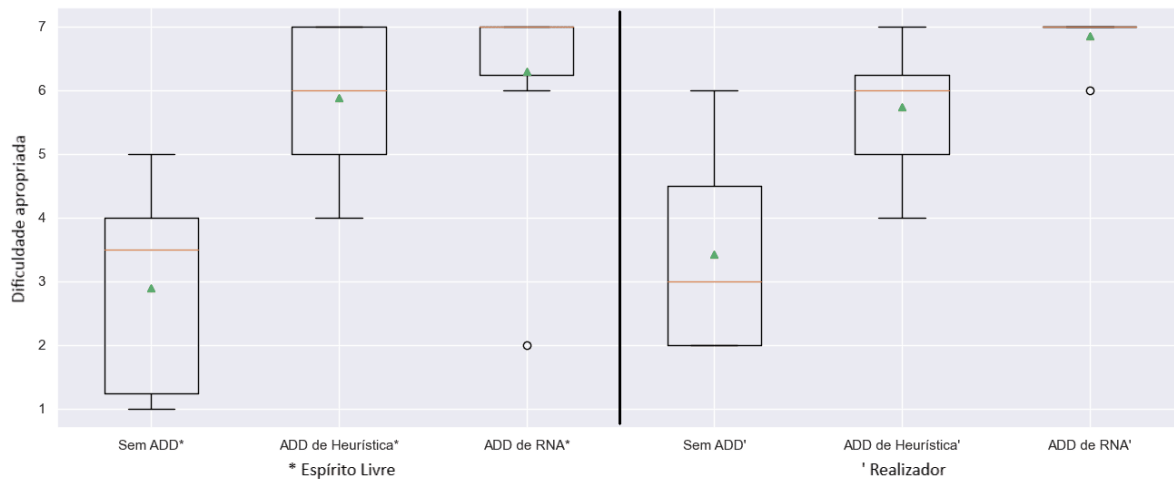


Figura 6.44: Diagrama de caixa das respostas referentes a dificuldade percebida no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.

As considerações apresentadas acima confirmam a Hipótese 7, com uma melhor adequação da dificuldade do jogo às habilidades do jogador quando utilizando métodos de Ajuste Dinâmico de Dificuldade. A adequação pode ser notada por estatísticas de desempenho dos jogadores, incluindo uma menor dispersão entre jogadores, e por média de respostas superiores em questionários aplicados aos jogadores. Ambos os modelos obtiveram sucesso em sua proposta, mas ocorreram resultados superiores com o Sistema de Análise de RNA.

Outras perguntas feitas aos participantes se referiam ao desafio proporcionado pelo jogo. Nesse sentido, as Figuras 6.45 e 6.46 apresentam as respostas referentes ao jogo ser desafiador, ao jogador conseguir superar os desafios mesmo se sentindo desafiado e sobre o jogador gostar da experiência de jogar o jogo. Os valores-p para cada uma dessas perguntas foram, respectivamente, de 0,359, 0,000004 e inferior a 0,000001.

O jogo foi mais desafiador para o jogo com ADD de RNA, seguido pelo sem ADD e com ADD de heurística. Essas médias foram próximas, mas são importantes quando associadas a resposta de conseguir superar os desafios. Ou seja, ainda que o jogo sem ADD seja desafiador, o jogador não consegue completar o que é desafiado (com média de 4,47). Por sua vez, mesmo com uma percepção de desafio maior com o ADD de RNA, os jogadores conseguem completar os desafios (6,65) mais do que quando aplicado o ADD de

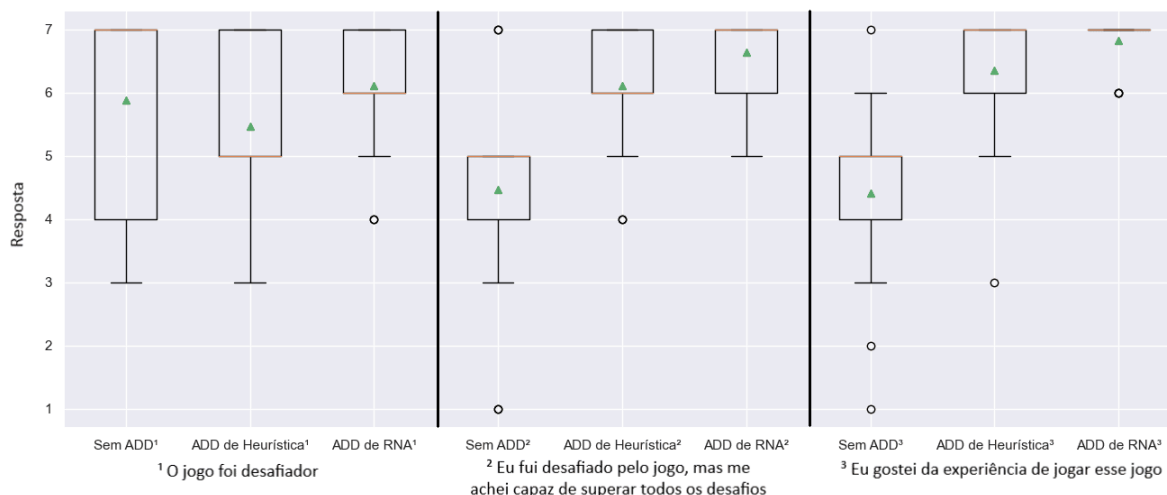


Figura 6.45: Diagrama de caixa das respostas referentes ao desafio proporcionado pelo jogo aos participantes na terceira bateria de testes, divididas por modelo de ADD.

heurísticas (6,12). Ademais, os participantes preferiram em média a experiência do jogo com ADD de RNA (6,83), seguido pelo de heurísticas (6,12) e sem ADD (4,42).

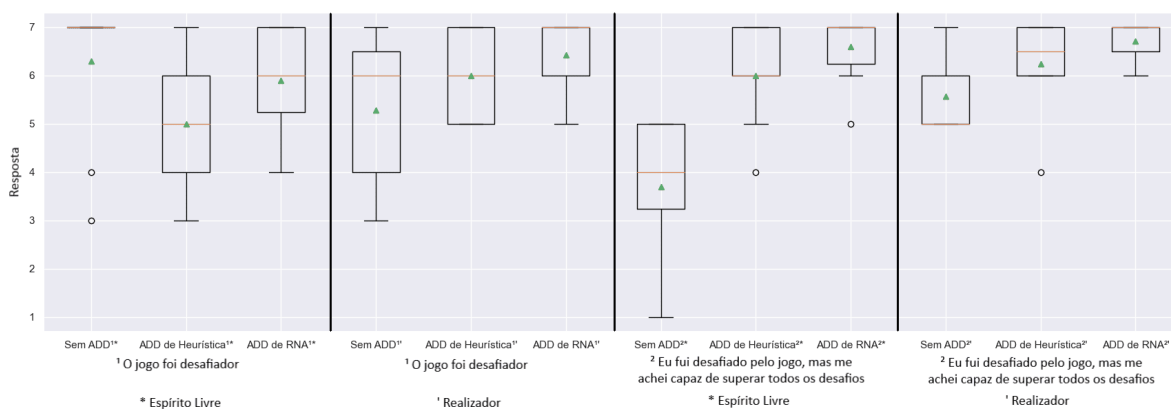


Figura 6.46: Diagrama de caixa das respostas referentes ao desafio proporcionado pelo jogo aos participantes na terceira bateria de testes, divididas por modelo de ADD e pelo perfil de jogador predominante entre Espírito Livre e Realizador.

Ao comparar perfis, jogadores de Espírito Livre consideram o jogo sem ADD mais desafiador, porém não conseguem superar esses desafios. Por outro lado, jogadores com predominância do perfil Realizador conseguem concluir os desafios, considerando que preferem geralmente maior dificuldade. Ao comparar os modelos de RNA e heurística, ocorreu variações similares independente do perfil, com médias superiores e menor dispersão para o ADD de RNA e para jogadores do perfil Realizador.

Como ilustrado na Figura 6.47, os participantes foram questionados sobre sentimentos experienciados no decorrer do jogo. O tédio e a frustração foram menores para ADD de RNA (com média de 2,12 e 2,24) e de heurísticas (2,12 e 2,54) em comparação a

sem ADD (4 e 5,36), com valores-p de 0,001274 e 0.000005, respectivamente. De modo complementar, a diversão foi maior em modelo com RNA (com média de 6,53) seguido de heurísticas (6,06) e sem ADD (4,12), com valor-p inferior a 0,000001.

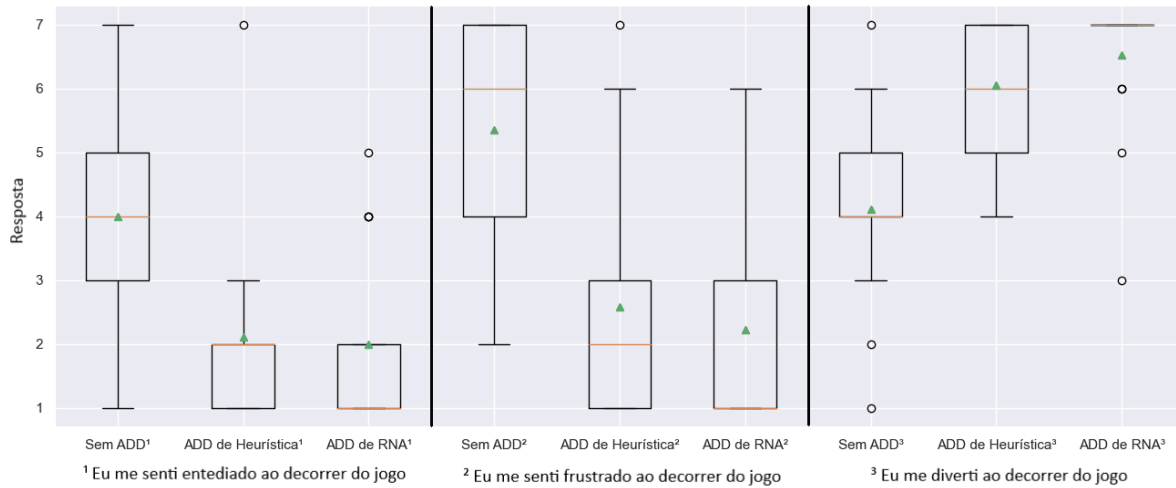


Figura 6.47: Diagrama de caixa das respostas referentes a sentimentos experienciados no jogo pelos participantes na terceira bateria de testes, divididas por modelo de ADD.

Como ilustrado na Figura 6.48, cinco perguntas foram feitas sobre as principais características de um jogador que se mantém no estado de fluxo durante uma sessão de jogo, com significância na diferença entre as médias ao apresentar valores-p abaixo de 0.000003 para as quatro primeiras perguntas e de 0.001547 para última. Nesse contexto, os participantes foram questionados se estavam focados no jogo, executavam ações automaticamente, perderam a noção do tempo, tinham controle de suas ações e estavam preocupados em falhar. As médias sem ADD foram piores em todas as perguntas em comparação aos distintos modelos de ADD, com os melhores resultados sendo para o modelo de RNA.

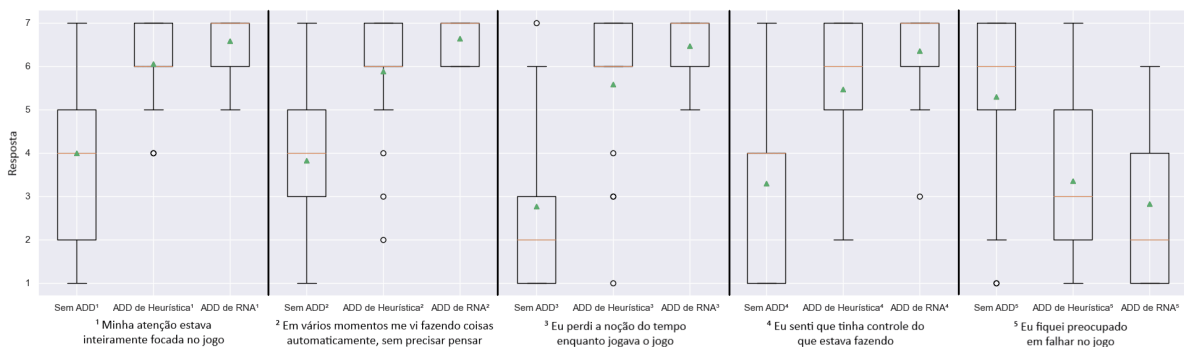


Figura 6.48: Diagrama de caixa das respostas referentes ao estado de fluxo experienciado pelos participantes na terceira bateria de testes, divididas por modelo de ADD.

Assim como especificado pela teoria do fluxo, as pessoas possuem diferentes zonas de fluxo de acordo com a habilidade, não havendo uma trajetória ideal de dificuldade única na atividade. Assim, o ADD variar de acordo com o perfil do jogador possibilita uma encaixe mais preciso do jogador em sua zona de fluxo.

Observa-se que há significância na diferença entre as médias para as perguntas do estado de fluxo, especialmente para o modelo de ADD com o Sistema de Análise por RNA. Desse modo, a Hipótese 8 é corroborada por manter os jogadores em estado de fluxo em um jogo com ADD de RNA adaptando o tamanho de plataformas e da amplitude do salto do avatar.

Capítulo 7

Considerações Finais

Este capítulo sintetiza o presente trabalho e as principais conclusões da pesquisa, além de levantar novos desafios para estudos futuros na área.

7.1 Principais contribuições

Os estudos de Ajuste Dinâmico de Dificuldade (ADD) estão crescendo em abrangência de temas e trabalhos realizados, considerando diversas variáveis como o desempenho, o estado afetivo e o perfil dos jogadores. Todavia, poucas pesquisas na literatura combinam esses fatores em um ajuste híbrido ou analisam partes específicas como parte de um sistema de jogo, como a análise e o controle de adaptações. Especificamente sobre o gênero plataforma, nenhuma pesquisa foi encontrada utilizando características dos perfis de jogadores para adaptação do jogo. Ademais, são escassas investigações aprofundadas sobre a melhoria da execução de ajustes ao considerar características do jogador além de sua habilidade, especialmente ao objetivar manter o jogador em estado de fluxo.

Por conseguinte, este trabalho objetivou propor melhorias à métodos de estimativa de dificuldade. Nesse sentido, foram propostas novas regras para definição da dificuldade, incluindo novas considerações para diferentes saltos no métodos de dificuldade de componentes de plataforma de Mourato. Também foram integrados outros componentes genéricos do gêneros de plataforma para dificuldade do nível, observando que características específicas dos componentes devem ser consideradas ao aplicar o método em distintos jogos, como as configurações disponíveis para inimigos ou o uso de distintos objetos de auxílio.

Foram definidas regras para criação de níveis ao testar de modelos de ADD, contendo a especificação do sistema de jogo entre Jogador, Monitoramento, Análise e Controle e devendo separar quais serão verificados para execução de testes. Para esse contexto,

também foram determinadas métricas para medição do estado de fluxo e do perfil de jogador, a partir da revisão da literatura e de análises das respostas dos participantes.

O foco esteve em adequar a dificuldade a cada jogador e mantê-lo em estado de fluxo ao investigar modelos de ADD em jogos de plataforma. Com isso, o jogo era modificado com base nos dados relativos previsão da dificuldade experienciada pelo jogador e a progressão geral da dificuldade era feita a partir da categorização do jogador entre os perfis Espírito Livre e Realizador do modelo Hexad.

Foram realizadas diversas adaptações no jogo de uso livre *The Explorer: 2D*, desenvolvido pela *Unity Technologies*. Empregaram-se alterações para criação de níveis, ajustes de componentes do jogo à proposta, comunicação com ferramentas externas utilizadas, disponibilização para múltiplos dispositivos e a opção de adaptar o jogo conforme os sistemas de um modelo de ADD:

1. Sistema de Controle:

- (a) Adaptação do tamanho de plataformas nos eixos horizontal e vertical.
- (b) Adaptação da amplitude do salto do avatar.
- (c) Adaptação combinada do tamanho de plataformas nos eixos horizontal e vertical e da amplitude do salto do avatar.

2. Sistema de Análise:

- (a) Análise dos componentes a serem ajustados a partir de heurísticas, considerando a dificuldade experienciada pelo jogador.
- (b) Análise dos componentes a serem ajustados a partir de método de aprendizado de máquina por rede neural direta, considerando a dificuldade experienciada e o perfil do jogador.

Para adaptação, avaliou-se a eficiência de métodos computacionais que estimassem a dificuldade e foi feita a seleção de atributos para previsão da dificuldade experienciada e do perfil do jogador. O método utilizado compreendia a dificuldade de um nível como o somatório da dificuldade de distintos componentes do gênero, obtidos pela probabilidade do sucesso de salto em uma plataforma ou por características intrínsecas e definidores de um componente.

A dificuldade experienciada foi analisada a partir de dados sobre a quantidade de mortes, a execução de saltos de sucesso por parte do jogador e o tempo jogado. Por outro lado, nota-se que jogadores do perfil Realizador buscam maestria, preferindo um maior desafio do jogo em comparação a jogadores do perfil Espírito Livre. Assim, o perfil do jogador entre Espírito Livre e Realizador foi analisado a partir de dados sobre a

quantidade de colecionáveis coletados, a execução de ataques a distância em comparação a ataques corpo-a-corpo e o tempo andando em direção ao final do nível, sem explorar por movimentações no sentido contrário. Enquanto os dados eram captados e analisados em tempo real, a adaptação era aplicada quando o jogador morria ou quando um nível era gerado.

7.2 Conclusões

Três baterias de testes foram conduzidas, com execução remota e uma rotina de apresentação da pesquisa e do contexto do jogo aos participantes. Jogadores com diferentes níveis de habilidade e familiaridade com jogos participaram dos experimentos. Ademais, dados sociodemográficos foram coletados previamente ao início do jogo e questionários foram aplicados ao final de cada nível e ao término da sessão, variando de acordo com o propósito do teste.

Pela análise de estudos anteriores e dos resultados apresentados na Seção 6.2, corroborou-se o uso do método de cálculo de dificuldade para associação com o ADD desenvolvido. Nesse cenário, foram incluídos casos adicionais para probabilidade de sucesso de um salto ao calcular a dificuldade de componentes de plataforma, como destinos com tamanho menor do que o destino final do salto, pontos de destino abaixo de pontos de origem e saltos em sentidos contrários à direção padrão de movimentação do jogo. Além disso, foram incluídas as dificuldade para demais componentes comuns aos jogos do gênero de plataforma, como inimigos, armadilhas e objetos de ajuda de movimento. Assim, observou-se uma relação entre a dificuldade estimada pelo algoritmo e experienciada pelos jogadores participantes do teste.

Também foi possível balancear o jogo em relação aos componentes utilizados e à definição de dados brutos que mais correlacionam à dificuldade experienciada e ao perfil dos jogadores. Desse modo, os níveis para teste do ADD na segunda e terceira baterias utilizaram dessas informações para definição dos níveis, sem a aplicação de condições que pudessem enviesar a medição da dificuldade e incluindo um planejamento dos níveis que não trouxesse desafio por quebra-cabeças ou armadilhas. Ademais, os níveis foram construídos com extremos de dificuldade para evitar a co-variância entre a dificuldade crescente dos níveis e a adaptação executada pelo ADD, o que impacta na percepção do estado de fluxo.

Pelos resultados expostos e analisados na Seção 6.3, constatou-se que um sistema de ADD é capaz de adequar a dificuldade e manter o jogador em estado de fluxo para um jogo do gênero plataforma. Em especial, a dificuldade foi constatada como mais adequada ao utilizar os modelos que só adaptam plataformas ou que combinam a adaptação de

plataformas e da amplitude do salto. Assim, demonstrou-se eficiência para diminuir a dispersão de desempenho entre os jogadores e melhorar as respostas informadas pelos participantes referentes ao estado de fluxo ou à adequação e progressão da dificuldade, com diferença significativa entre as médias.

Por sua vez, pelos resultados expostos e analisados na Seção 6.4, constatou-se que um sistema de ADD com análise executada por rede neural direta é capaz de adequar a dificuldade e manter o jogador em estado de fluxo para um jogo do gênero plataforma ao aplicar características do perfil de jogador para progressão de dificuldade. Nesse contexto, foi observada diminuição na dispersão de desempenho entre os jogadores e melhora nas respostas informadas pelos participantes referentes ao estado de fluxo ou à adequação e progressão da dificuldade, com diferença significativa entre as médias.

Verificou-se que o desempenho dos jogadores consegue ser superior nos casos com a aplicação de ADD, incluindo uma maior frequência de jogadores completando níveis. Ademais, houve menor frustração, menor tédio, maior diversão e melhor percepção de desempenho próprio nos casos com aplicação de ADD. Obteve-se a validação da adequação da dificuldade e do estado de fluxo para ambos os modelos de ADD, ocorrendo melhores resultados para o modelo com análise por rede neural e considerando o perfil do jogador.

7.3 Trabalhos futuros

A continuidade de estudos referentes ao estado afetivo constitui um caminho para trabalhos futuros verificando de modo mais aprofundado o cálculo de determinação do estado afetivo do jogador e analisando a associação de EDA com outros sinais no contexto do jogo. Técnicas de aprendizado de máquina também podem ser empregadas para análise do estado afetivo. Ademais, este trabalho utilizou-se de redes neurais diretas para prever o perfil de jogador e a dificuldade experienciada, porém outros modelos de redes neurais e demais métodos de aprendizado de máquina podem ser testados e comparados.

Em relação ao Ajuste Dinâmico de Dificuldade empregado, constitui como objeto de estudo delineamentos mais simples e focados de características e componentes do jogo. Assim, pode ser examinada a adaptação de outros componentes separadamente ou em conjunto, como inimigos e objetos de auxílio, e de características da jogabilidade, como a presença de um esquema de vidas. Além disso, o momento que o ajuste é empregado é um ponto a ser averiguado, considerando que pode influenciar na percepção dos ajustes por parte do jogador e na experiência geral. Alguns momentos possíveis são: ao criar níveis, ao morrer e em tempo real.

Pode-se analisar a dificuldade como influenciada pelo *design* do jogo e não somente pela presença e pelas características de componentes. Assim, conseguiria ser averiguado o uso

de armadilhas e de quebra-cabeças, além da presença de missões e conteúdos específicos para desafiar o jogador. Nesse cenário, torna-se possível a integração do ADD com a Geração de Conteúdo Procedural (GCP), associando à criação de geometria e de desafios que se adéquem ao jogador.

Por fim, a adaptatividade pode ser aplicada no contexto de saúde dos jogadores. Nesse sentido, é possível integrar o ADD em estudos como uma mecânica auxiliadora na reabilitação de membros motores em jogos envolvendo ferramentas com movimentação, no treinamento de memória ou na redução de ansiedade.

Referências

- [1] Mourato, Fausto J. S. V. e Manuel P. Santos: *Measuring difficulty in platform videogames*. Em *4^a Conferência Nacional em Interação Pessoa-Máquina*, 2010. xi, 3, 4, 8, 29, 46, 50, 52, 104
- [2] Russell, James A.: *A circumplex model of affect*. *Journal of Personality and Social Psychology*, 1980. xi, 11
- [3] Marczewski, Andrzej: *User types*. Even Ninja Monkeys Like to Play: Gamification, Game Thinking, and Motivational Design, páginas 65–80, 2015. xi, 14, 110, 118, 119
- [4] Chen, Jenova: *Flow in games (and everything else)*. *ACM magazine*, 50(4), 2007. xi, 1, 16, 17, 21, 22, 57
- [5] Bontchev, Boyan: *Adaptation in affective video games: A literature review*. *Cybernetics and Information Technologies*, 16(3):3–34, 2016. xi, 1, 21, 23, 41, 57
- [6] Sorenson, Nathan e Philippe Pasquier: *Towards a generic framework for automated video game level creation*. *Applications of Evolutionary Computation*, 2010. xi, 24
- [7] Smith, Gillian, Mee Cha e Jim Whitehead: *A framework for analysis of 2d platformer levels*. *Sandbox '08 Proceedings of the 2008 ACM SIGGRAPH symposium on Video games*, páginas 75–80, 2008. xi, 23, 24, 25, 57, 102, 104
- [8] Koens, Erik: *Generating non-monotone 2D platform levels and predicting difficulty*. *Dissertação de mestrado*, Utrecht University, 2015. xi, 3, 29, 30, 121
- [9] Blom, Paris, Sander Bakkes e Pieter Spronck: *Modeling and adjusting in-game difficulty based on facial expression analysis*. *Entertainment Computing*, 31:100307, julho 2019. xi, 31, 32
- [10] Araujo, Vladimir, Diego Mendez e Alejandra Gonzalez: *A novel approach to working memory training based on robotics and ai*. *Information*, 10:350, novembro 2019. xi, 32, 33
- [11] Hawkins, Guy, Keith Nesbitt e Scott D. Brown: *Dynamic difficulty balancing for cautious players and risk takers*. *Int. J. Computer Games Technology*, páginas 1–10, 2012. xi, 2, 14, 39, 40, 77, 91, 109

- [12] Mourato, Fausto J. S. V.: *Enhancing Automatic Level Generation for Platform Videogames*. Tese de doutorado, Universidade Nova de Lisboa, 2015. xi, xii, xix, 3, 7, 8, 9, 46, 47, 50, 51, 54, 55, 57, 121
- [13] Rosa, Marcos P. C.: *Ajuste dinâmico de dificuldade híbrido em um jogo do gênero plataforma*. Dissertação de Graduação, Universidade de Brasília, 2019. xii, xiii, xix, 2, 3, 49, 52, 55, 69, 70, 75, 76, 78, 79, 82, 83, 85, 90, 118, 126
- [14] Gularte, Daniel: *Jogos eletrônicos: 50 anos de interação e diversão*. 2AB Editora, 2010, ISBN 8560284184, 9788560284184. xix, 7
- [15] Lopes, Ricardo e Rafael Bidarra: *Adaptivity challenges in games and simulations: A survey*. IEEE Transactions on Computational Intelligence and AI in Games, páginas 85–99, 2011. 1, 19
- [16] Novak, Jeannie: *Desenvolvimento de Games*. Cengage Learning, 2011, ISBN 9788522106325. 1, 9, 19
- [17] Schell, Jesse: *The Art of Game Design: A Book of Lenses*. AK Peters, 2ª edição, 2014, ISBN 1466598646, 9781466598645. 1, 21
- [18] Chang, David M. J.: *Dynamic difficulty adjustment in computer games*. Proceedings of the 11th Annual Interactive Multimedia Systems Conference, 2013. 1, 21
- [19] Andrade, Gustavo, Geber Ramalho, Alex Sandro Gomes e Vincent Corruble: *Dynamic game balancing: an evaluation of user satisfaction*. American Association for Artificial Intelligence, 2006. 1, 21
- [20] Araujo, Bruno B. P. L.: *Um estudo sobre adaptatividade dinâmica de dificuldade em jogos*. Dissertação de mestrado, Pontifícia Universidade Católica do Rio de Janeiro, 2012. 2, 18, 19, 20, 22, 37
- [21] Sha, Lingdao: *Creating appropriate challenge level game opponent by the use of dynamic difficulty adjustment*. Sixth International Conference on Natural Computation, 8:3897–3901, 2010. 2
- [22] Denisova, Alena e Paul Cairns: *Adaptation in digital games: The effect of challenge adjustment on player performance and experience*. Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play, páginas 97–101, 2015. 2, 39
- [23] Hintze, Arend, Randal Olson e Joel Lehman: *Orthogonally evolved ai to improve difficulty adjustment in video games*. Applications of Evolutionary Computation, páginas 525–540, 2016. 2, 38
- [24] 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. 2, 38
- [25] Liu, Changchun, Pramila Agrawal, Nilanjan Sarkar e Shuo Chen: *Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback*. International Journal of Human-Computer Interaction, páginas 506–529, 2009. 2, 42

- [26] Chanel, Guillaume, Cyril Rebetez, Mireille Bétrancourt e Thierry Pun: *Emotion assessment from physiological signals for adaptation of game difficulty*. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2011. 2, 11
- [27] Imre, Daniel: *Real-time Analysis of Skin Conductance for Affective Dynamic Difficulty Adjustment in Video Games*. Tese de doutorado, Algoma University, 2016. 2, 42
- [28] Vollmers, Casper Gade Kranker: *Dynamic Difficulty Adjustment In Games Using Physiology*. Dissertação de mestrado, Aalborg University Copenhagen, 2018. 2, 48
- [29] Yu, Xinrui, Suoju He, Yuan Gao, Jiajian Yang, Lingdao Sha, Yidan Zhang e Zhaobo Ai: *Dynamic difficulty adjustment of game ai for video game dead-end*. The 3rd International Conference on Information Sciences and Interaction Sciences, junho 2010. 2, 91
- [30] Santos, Eduardo e Iago Moraes: *Comparação de ajustes dinâmicos de dificuldade aplicados a diferentes elementos de jogo*. Dissertação de Graduação, Universidade de Brasília, 2021. 2, 4, 49, 50, 65, 69, 75, 77, 90, 91, 118, 126
- [31] Stenros, Jakko: *The game definition game: A review*. Games and Culture, 12(6):499–520, 2016. 5
- [32] 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. 5, 19
- [33] Huizinga, Johan: *Homo Ludens: o jogo como elemento da cultura*. Perspectiva, 6ª edição, 2010, ISBN 8527300753, 9788527300759. 6, 9, 17
- [34] Koster, Raph: *A Theory of Fun for Game Design*. Paraglyph Press, 1ª edição, 2004, ISBN 1449363210, 9781449363215. 6, 9
- [35] Salen, Katie e Eric Zimmerman: *Rules of Play: Game Design Fundamentals*. The MIT Press, 2003, ISBN 0262240459, 9780262240451. 6, 9, 10
- [36] Salen, Katie e Eric Zimmerman: *The Game Design Reader: A Rules of Play Anthology*. The MIT Press, janeiro 2006, ISBN 0262195364, 9780262195362. 6
- [37] Prado, Gilbertto e Hélia Vannucchi: *Discutindo o conceito de gameplay*. Texto Digital, páginas 130–140, 2009. 6
- [38] Xavier, Guilherme: *Condição Eletrolúdica: Cultura Visual Nos Jogos Eletrônicos*. Editora Novas Idéias, 2010, ISBN 9788560284153. 6
- [39] Mello, Vinícius e Letícia Perani: *Gameplay x playability: defining concepts, tracing differences*. SBC - Proceedings of SBGames 2012, 2012. 6
- [40] Crawford, Chris: *The art of computer game design*. Osborne/McGraw-Hill Berkley, 1984, ISBN 0881341177. 6

- [41] Minkkinen, Toni: *Basics of Platform Games*. Dissertação de graduação, Kajaanin Ammattikorkeakoulu University of Applied Sciences, 2016. 7, 67
- [42] Fullerton, Tracy: *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. A K Peters/CRC Press, 3ª edição, 2014, ISBN 1482217163, 9781482217162. 9
- [43] Aguiar, João V. S. e Lucas V. Fernandes: *Ajuste dinâmico de dificuldade em jogos a partir de variáveis do jogo e do usuário*. Dissertação de graduação, Universidade de Brasília, 2017. 9, 18, 22, 43, 49
- [44] Fortugno, Nick: *The strange case of the casual gamer*. Game Usability: Advice from the Experts for Advancing the Player Experience, 2008. 10
- [45] Broscha, Tobias, Klaus R. Scherer, Didier Grandjean e David Sanders: *The impact of emotion on perception, attention, memory, and decision-making*. Swiss Med Weekly, 2013. 10
- [46] Chang, Luke J. e Alec Smith: *Social emotions and psychological games*. Behavioral Sciences, páginas 133–140, 2015. 10
- [47] Lang, Peter J.: *The emotion probe: studies of motivation and attention*. American psychologist, 1995. 10, 11
- [48] Myers, David G.: *Psicologia*. LTC, 9ª edição, 2015, ISBN 9788521620372. 11
- [49] Soares, Rodrigo T.: *Biofeedback sensors in game telemetry research*. SBC – Proceedings of SBGames 2016, 2016. 12
- [50] Soares, Rodrigo T.: *Biofeedback sensors in electronic games: A practical evaluation*. SBC – Proceedings of SBGames 2017, 2017. 12
- [51] Csikszentmihalyi, Mihaly: *Flow: The Psychology of Optimal Experience*. New York: Harper & Row, 1992, ISBN 0061339202, 9780061339202. 12, 15, 16, 37
- [52] John Hallam, Georgios N. Yannakakis e: *Real-time game adaptation for optimizing player satisfaction*. IEEE Transactions on Computational Intelligence and AI in Games, página 121–133, 2012. 13, 35
- [53] Bartle, Richard: *Hearts, clubs, diamonds, spades: Players who suit muds*. Journal of MUD research, 1(1):19, junho 1996. 13
- [54] Aleksieva-Petrova, A, M Petrov e B Bontchev: *Game and learner ontology model*. Em *Int. Scientific Conf. Computer Science'2011*, páginas 1–2, 2011. 13
- [55] Kolb, David: *Experiential Learning: Experience As The Source Of Learning And Development*, volume 1. Journal of Business Ethics, janeiro 1984, ISBN 0132952610. 13
- [56] Honey, Peter e Alan Mumford: *The manual of learning styles*, volume 3. Peter Honey Publications, janeiro 1992, ISBN 0950844470. 13

- [57] Tondello, Gustavo, Rina Wehbe, Lisa Diamond, Marc Busch, Andrzej Marczewski e Lennart Nacke: *The gamification user types hexad scale*. Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play, páginas 229–243, outubro 2016. 14, 110, 111
- [58] John, Oliver P. e Sanjay Srivastava: *The big five trait taxonomy: History, measurement, and theoretical perspectives*. Em *Handbook of Personality: Theory and Research*, páginas 102–138, 1999. 14, 40
- [59] Cowley, Ben, Darryl Charles, Michaela Black e Ray Hickey: *Toward an understanding of flow in video games*. Computers in Entertainment, 2008. 16
- [60] Sherry, John L.: *Flow and media enjoyment*. Communication Theory, páginas 328–347, 2004. 16
- [61] Sweetser, Penelope e Peta Wyeth: *Gameflow: A model for evaluating player enjoyment in games*. Computers in Entertainment, 3, julho 2005. 17
- [62] Aponte, Maria Virginia e Guillaume Levieux: *Difficulty in video games : An experimental validation of a formal definition*. Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology, 2011. 17, 18, 50
- [63] Byrne, Ed: *Game Level Design*. Charles River Media, 1ª edição, 2004, ISBN 1584503696, 978-1584503699. 17
- [64] Parnandi, Avinash e Ricardo Gutierrez-Osuna: *A comparative study of game mechanics and control laws for an adaptive physiological game*. Journal on Multimodal User Interfaces, 9(1):31–42, março 2015. 20
- [65] Fairclough, Stephen H.: *Fundamentals of physiological computing*. Interacting with Computers, 21(1-2):133–145, janeiro 2009. 20
- [66] Compton, Kate e Michael Mateas: *Procedural level design for platform games*. American Association for Artificial Intelligence, 2006. 23, 24, 30
- [67] Bundy, Alan: *Preparing for the future of artificial intelligence*. AI & SOCIETY, 32, novembro 2016. 25
- [68] Danks, David e Alex London: *Algorithmic bias in autonomous systems*. Em *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*, páginas 4691–4697, agosto 2017. 25
- [69] Ferràs-Hernández, Xavier: *The future of management in a world of electronic brains*. Journal of Management Inquiry, 27:105649261772497, agosto 2017. 25
- [70] Brynjolfsson, Erik e Tom Mitchell: *What can machine learning do? workforce implications*. Science, 358:1530–1534, dezembro 2017. 25
- [71] Hammedi, Sanda, Fathi Essalmi, Mohamed Jemni e Alaa Qaffas: *An investigation of ai in games: educational intelligent games vs non-educational games*. Em *2020 International Multi-Conference on: “Organization of Knowledge and Advanced Technologies” (OCTA)*, páginas 1–4, fevereiro 2020. 26, 31

- [72] Ishihara, Makoto, Suguru Ito, Ryota Ishii, Tomohiro Harada e Ruck Thawonmas: *Monte-carlo tree search for implementation of dynamic difficulty adjustment fighting game ais having believable behaviors*. Em *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, páginas 1–8, agosto 2018. 26, 34
- [73] McGlinchey, Stephen e Darryl Charles: *The past, present and future of artificial neural networks in digital games*. Proceedings of the 5th international conference on computer games: artificial intelligence, design and education, página 163–169, 2004. 26
- [74] Galway, Leo, Darryl Charles e Michaela Black: *Machine learning in digital games: A survey*. Artificial Intelligence Review, 29:123–161, abril 2008. 26
- [75] McGlinchey, Stephen: *Learning of ai players from game observation data*. 4th International Conference on Intelligent Games and Simulation, páginas 106–110, 2003. 27
- [76] Wehbe, Rina R., Elisa D. Mekler, Mike Schaeckermann, Edward Lank e Lennart E. Nacke: *Testing incremental difficulty design in platformer games*. Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, páginas 5109–5113, 2017. 28
- [77] Silva, Mirna, Victor Silva e Luiz Chaimowicz: *Dynamic difficulty adjustment on moba games*. Entertainment Computing, 18, 2017. 31
- [78] Demediuk, Simon, Marco Tamassia, William Raffe, Fabio Zambetta, Xiaodong Li e Florian Mueller: *Monte carlo tree search based algorithms for dynamic difficulty adjustment*. Em *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, páginas 53–59, agosto 2017. 34
- [79] Demediuk, Simon, Marco Tamassia, William Raffe, Fabio Zambetta, Florian Mueller e Xiaodong Li: *Measuring player skill using dynamic difficulty adjustment*. Em *ACSW '18: Proceedings of the Australasian Computer Science Week Multiconference*, janeiro 2018. 34
- [80] Kusano, Takahiro, Yunshi Liu, Pujana Paliyawan, Tomohiro Harada e Ruck Thawonmas: *Motion gaming ai using time series forecasting and dynamic difficulty adjustment for improving exercise balance and enjoyment*. Em *IEEE Conference on Games 2019 (CoG 2019)*, agosto 2019. 34
- [81] Fujita, Kazuhisa: *Alphadda: Game artificial intelligence with dynamic difficulty adjustment using alphazero*. ArXiv, abs/2111.06266, novembro 2021. 35
- [82] Or, Dvir Ben, Michael Kolomenkin e Gil Shabat: *Dl-dda - deep learning based dynamic difficulty adjustment with ux and gameplay constraints*. Em *2021 IEEE Conference on Games (CoG)*, páginas 1–7, 2021. 35
- [83] Zohaib, Mohammad: *Dynamic difficulty adjustment (dda) in computer games: A review*. Advances in Human-Computer Interaction, páginas 1–12, novembro 2018. 36

- [84] Calvillo-Gámez, Eduardo H., Paul A. Cairns e Anna Louise Cox: *Assessing the core elements of the gaming experience*. Evaluating User Experience in Games, 1:47–71, 2010. 37
- [85] Charles, Darryl e Michaela Black: *Dynamic player modelling: A framework for player-centred digital games*. Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education, páginas 29–35, janeiro 2004. 37
- [86] Christyowidiasmoro, Ramadhany Candra Arif Putra e Supeno Mardi Susiki: *Measuring level of difficulty in game using challenging rate (cr) on 2d real time strategy line defense game*. 2015 International Electronics Symposium, página 218–222, 2015. 38
- [87] Nagle, Aniket, Peter Wolf e Robert Riener: *Towards a system of customized video game mechanics based on player personality: Relating the big five personality traits with difficulty adaptation in a first-person shooter game*. Entertainment Computing, 13:10–24, 2016, ISSN 1875-9521. 40
- [88] Moschovitis, Paraschos e Alena Denisova: *Keep calm and aim for the head: Biofeedback-controlled dynamic difficulty adjustment in a horror game*. IEEE Transactions on Games, páginas 1–1, 2022. 43
- [89] Natal, Guilherme V. R.: *Uso de sensores no ajuste dinâmico de dificuldade híbrido em jogos*. Dissertação de graduação, Universidade de Brasília, 2018. 43, 49
- [90] Pedersen, C., J. Togelius e G. N. Yannakakis: *Modeling player experience in super mario bros*. Proceedings of the 2009 IEEE Symposium on Computational Intelligence and Games (CIG), página 132–139, setembro 2009. 44, 45
- [91] Yannakakis, G. N. e J. Hallam: *Game and player feature selection for entertainment capture*. Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Games, página 244–251, abril 2007. 44
- [92] Pedersen, C., J. Togelius e G. N. Yannakakis: *Modeling player experience for content creation*. IEEE Transactions on Computational Intelligence and AI in Games, 2(1):54–67, 2010. 45
- [93] Shaker, N., G. Yannakakis e J. Togelius: *Towards automatic personalized content generation for platform games*. Proceedings of the 6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE), página 63–68, outubro 2010. 45
- [94] Jennings-Teats, Martin, Gillian Smith e Noah Wardrip-Fruin: *Polymorph: Dynamic difficulty adjustment through level generation*. Proceedings of the 2010 Workshop on Procedural Content Generation in Games, 2010. 45
- [95] Jennings-Teats, Martin, Gillian Smith e Noah Wardrip-Fruin: *Polymorph: A model for dynamic level generation*. Proceedings of the 6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE), página 138–143, October 2010. 45

- [96] Mourato, Fausto J. S. V., Fernando Birra e Manuel P. Santos: *Difficulty in action based challenges: Success prediction, players' strategies and profiling*. Proceedings of the 11th Conference on Advances in Computer Entertainment Technology, páginas 1–10, 2014. 46, 54, 101
- [97] Bakkes, S., S. Whiteson, G. Li, G. V. Vişniuc, E. Charitos, N. Heijne e A. Swellegrebel: *Challenge balancing for personalised game spaces*. 2014 IEEE Games Media Entertainment, páginas 1–8, 2014. 47
- [98] Shi, P. e K. Chen: *Learning constructive primitives for real-time dynamic difficulty adjustment in super mario bros*. IEEE Transactions on Games, 10(2):155–169, 2018. 47
- [99] Parekh, Ravi: *Staying in the Flow using Procedural Content Generation and Dynamic Difficulty Adjustment*. Dissertação de mestrado, Worcester Polytechnic Institute, 2017. 47
- [100] Wheat, Daniel: *Dynamically adjusting game-play in 2D platformers using procedural level generation*. Dissertação de graduação, Edith Cowan University, 2013. 48, 57
- [101] Greco, Alberto, Gaetano Valenza e Enzo Scilingo: *Modeling for the Analysis of the EDA*. Advances in Electrodermal Activity Processing with Applications for Mental Health, novembro 2016, ISBN 978-3-319-46704-7. 56
- [102] Dahlskog, Steve e Julian Togelius: *Procedural content generation using patterns as objectives*. Applications of Evolutionary Computation, páginas 325–336, 2014. 57
- [103] Rosa, Marcos, Eduardo Santos, Iago Moraes, Tiago Silva, Mauricio Sarmet, Carla Castanho e Ricardo Jacobi: *Dynamic difficulty adjustment using performance and affective data in a platform game*. Em *HCI International 2021 - Late Breaking Papers: Design and User Experience*, páginas 367–386, Cham, 2021. Springer International Publishing, ISBN 978-3-030-90238-4. 75
- [104] Diamond, Lisa, Gustavo Tondello, Andrzej Marczewski, Lennart Nacke e Manfred Tscheligi: *The hexad gamification user types questionnaire: Background and development process*. Workshop on Personalization in Serious and Persuasive Games and Gamified Interactions, outubro 2015. 93
- [105] Krath, Jeanine e Harald von Korflesch: *Player types and game element preferences: Investigating the relationship with the gamification user types hexad scale*. Em *HCI in Games: Experience Design and Game Mechanics*, páginas 219–238, Cham, 2021. Springer International Publishing, ISBN 978-3-030-77277-2. 93, 110
- [106] Santos, Ana, Wilk Oliveira, Maximilian Altmeyer, Juho Hamari e Seiji Isotani: *Psychometric investigation of the gamification hexad user types scale in brazilian portuguese*. Scientific Reports, 12, março 2022. 93, 110
- [107] Fang, Xiaowen, Jingli Zhang e Susy S. Chan: *Development of an instrument for studying flow in computer game play*. International Journal of Human-Computer Interaction, 29(7):456–470, 2013. 93

- [108] Jackson, Susan e Herbert Marsh: *Development and validation of a scale to measure optimal experience: The flow state scale*. Journal of Sport and Exercise Psychology, 18:17–35, 1996. 93
- [109] Kyriazos, T., A. Stalikas, K. Prassa, M. Galanakis, K. Flora e V. Chatzilia: *The flow short scale (fss) dimensionality and what mimic shows on heterogeneity and invariance*. Psychology, 9:1357–1382, 2018. 93
- [110] Heaton, Jeffrey: *The number of hidden layers*. Heaton Research Inc, janeiro 2008. 112

Apêndice A

Questionário

Perguntas sobre Dados Sociodemográficos foram aplicadas antes do jogo iniciar, perguntas sobre o Nível ao final de cada nível e perguntas do 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 7]
 - Casual - Dedicado
4. Você está familiarizado com jogos de plataforma (ex: Mario, Sonic)? [escala de 1 a 7]
 - Pouco familiarizado - Muito familiarizado
5. Qual a sua relação com jogos de plataforma? [escala de 1 a 7]

- Não gosta - Gosta muito
6. Qual a sua dificuldade preferida em jogos? [múltipla escolha]
- Fácil
 - Médio
 - Difícil
7. Com que frequência você joga nos seguintes dispositivos? [grade de escalas de 1 a 7]
- Nunca - Diariamente
- (a) Celular / tablet
- (b) Computador pessoal (desktop ou laptop)
- (c) Outros (ex: Playstation, Xbox)
8. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
- Discordo muito - Concordo muito
- (a) É importante para mim seguir o meu próprio caminho
- (b) Eu me vejo buscando situações de alto-risco frequentemente
- (c) Eu gosto de provocar situações inesperadas
- (d) Eu sou muito ambicioso
- (e) Eu sempre opto pelo caminho mais seguro
- (f) Eu gosto de competições onde um prêmio pode ser ganho
- (g) Muitas vezes deixo minha curiosidade me guiar
- (h) Gosto de questionar o status quo
- (i) As recompensas são uma ótima maneira de me motivar
- (j) Eu desisto facilmente
- (k) Eu gosto de experimentar coisas novas
- (l) Gosto de vencer obstáculos
- (m) Eu cuido dos meus próprios interesses
- (n) Eu penso muito antes de agir
- (o) Eu me vejo como um rebelde

- (p) É importante para mim sempre realizar minhas tarefas completamente
- (q) Prefiro definir meus próprios objetivos
- (r) Não gosto de seguir regras
- (s) É difícil para mim deixar de lado um problema antes de encontrar uma solução
- (t) Situações de risco me deixam empolgado
- (u) O retorno do investimento é importante para mim
- (v) Ser independente é importante para mim
- (w) Gosto de dominar tarefas difíceis
- (x) Eu gosto de mudar as coisas com minhas próprias mãos
- (y) Eu sempre alcanço meus objetivos não importando a dificuldade
- (z) Se a recompensa for suficiente, vou me esforçar

A.2 Primeira bateria de testes

A.2.1 Nível

1. Como você descreveria a dificuldade do nível? [escala de 0 a 10]
 - Muito fácil - Muito difícil
2. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
 - Discordo muito - Concordo muito
 - (a) Eu me diverti com o nível
 - (b) Eu me senti desafiado pelo nível
 - (c) Eu me senti frustrado jogando o nível
 - (d) Eu me senti entediado jogando o nível

A.2.2 Jogo

1. Como você considera o seu desempenho no jogo? [escala de 1 a 7]
 - Péssimo - Ótimo
2. Em relação a cada opção, você concorda que as seguintes ações foram importantes para influenciar o seu desempenho? [escala de 1 a 7]

- Alcançar o final do nível
 - Coletar chaves (coleccionáveis amarelos)
 - Dominar os controles da protagonista
 - Completar o nível o mais rápido possível
 - Derrotar o máximo de inimigos possível
 - Explorar ao máximo o ambiente
 - Melhorar o desempenho próprio no jogo
3. Em relação a cada opção, o jogo se tornou mais desafiador com a presença deste elemento? [grade de escalas de 1 a 7]
- Discordo muito - Concordo muito
- (a) Plataformas que se movem
- (b) Plataformas distantes (precisa pular mais longe para alcançar)
- (c) Tamanho reduzido das plataformas (espaço menor para andar)
4. Em relação a cada opção, o jogo se tornou mais desafiador com a presença deste elemento? [grade de escalas de 1 a 7]
- Discordo muito - Concordo muito
- (a) Obstáculos (objetos que pode destruir ou empurrar)
- (b) Espinhos
- (c) Inimigos com ataque corpo-a-corpo (inimigo rosa)
- (d) Inimigos com ataque à distância (inimigo azul)
5. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
- Discordo muito - Concordo muito
- (a) Eu me arrisquei durante o jogo
- (b) Eu procurei completar o jogo sem desistir

A.3 Segunda e terceira bateria de testes

A.3.1 Jogo

1. Como você considera o seu desempenho no jogo? [escala de 1 a 7]
 - Péssimo - Ótimo

2. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
 - Discordo muito - Concordo muito

 - (a) Eu me senti entediado no decorrer do jogo
 - (b) Eu me senti frustrado no decorrer do jogo
 - (c) Eu me diverti no decorrer do jogo
 - (d) A dificuldade do jogo foi apropriada para as minhas habilidades
 - (e) Eu senti que a dificuldade do jogo era discrepante ao longo dos níveis

3. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
 - Discordo muito - Concordo muito

 - (a) Minha atenção estava inteiramente focada no jogo
 - (b) Em vários momentos me vi fazendo coisas automaticamente, sem precisar pensar
 - (c) Eu perdi a noção do tempo enquanto jogava o jogo
 - (d) Eu senti que tinha controle do que estava fazendo
 - (e) Eu fiquei preocupado em falhar no jogo

4. Você concorda com a seguinte afirmação? [grade de escalas de 1 a 7]
 - Discordo muito - Concordo muito

 - (a) Percebi alterações no tamanho de plataformas
 - (b) Percebi alterações no salto da personagem
 - (c) A percepção de alterações foi negativa para minha experiência de jogo