

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

LANA BERTOLDO ROSSATO

**Design Evolutivo de Jogos de Mesa: Um
Estudo de Caso no Jogo Risk**

Dissertação apresentada como requisito parcial
para a obtenção do grau de Mestre em Ciência da
Computação

Orientador: Prof. Dr. Anderson Rocha Tavares

Porto Alegre
2023

CIP — CATALOGAÇÃO NA PUBLICAÇÃO

Rossato, Lana Bertoldo

Design Evolutivo de Jogos de Mesa: Um Estudo de Caso no Jogo Risk / Lana Bertoldo Rossato. – Porto Alegre: PPGC da UFRGS, 2023.

78 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2023. Orientador: Anderson Rocha Tavares.

1. *Evolutionary Game Design*. 2. Algoritmo Genético. 3. Risk. I. Tavares, Anderson Rocha. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões

Vice-Reitora: Prof^a. Patricia Pranke

Pró-Reitor de Pós-Graduação: Prof. Celso Giannetti Loureiro Chaves

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenador do PPGC: Prof. Claudio Rosito Jung

Bibliotecário-chefe do Instituto de Informática: Alexsander Borges Ribeiro

*“We’ve tended to forget that no computer
will ever ask a new question.”*

— GRACE HOPPER

AGRADECIMENTOS

Esses anos de mestrado foram muito intensos e tenho o orgulho de dizer que consegui passar por tudo de cabeça erguida.

Acima de tudo, agradeço aos meus pais, Volnei e Marisa, que não mediram esforços para me ajudar a seguir a carreira que tanto quis, mesmo às vezes sem saber ao certo que carreira era essa. Sempre entenderam o meu lado quando dizia com tristeza que não conseguiria viajar para matar a saudade porque tinha trabalho atrasado ou tinha que cuidar dos milhares de testes que estavam rodando em vários computadores. Eu não seria nada sem vocês.

Agradeço também aos meus orientadores que me fizeram ser quem eu sou nesse mundo acadêmico. Em especial ao Prof. Anderson que, com toda a sua calma e dedicação e mesmo sem saber, me ajudou a sair do lugar mais escuro que já estive e me fez recuperar a vontade de ir para a universidade e fazer pesquisa. Para alguém que não queria nem terminar o mestrado, estar pronta para enfrentar o doutorado é motivo de muita comemoração.

Não posso deixar de agradecer à minha psicóloga Letícia por me encher de força e me mostrar que sou capaz, mesmo tendo dúvidas a todo momento. Esse agradecimento pode ser estranho, mas sem ela eu não entenderia metade das coisas que entendo agora e não teria forças nem para sair da cama.

Agradeço às minhas amigas Moniele e Talita por serem meu porto seguro nessa cidade nova e embarcarem comigo nas mais diversas aventuras, desde visitar apartamentos até ir na UBS para tirar um piercing que deu errado. Vocês são muito importantes para mim e com certeza fazem parte da fase mais louca da minha vida. Agradeço também ao meu amigo Gabriel que, mesmo distante, me ajudou no dia a dia a minimizar o estresse e a pressão que eu mesma coloco em mim.

Também gostaria de agradecer aos professores e colegas do Logics, laboratório que me acolheu. Mesmo sendo de uma área totalmente diferente, nunca me senti tão pertencente a um lugar. Achei que a pós-graduação fosse um lugar competitivo, mas vocês me mostraram que é totalmente ao contrário. Sempre me ajudaram e me incentivaram, e isso eu jamais vou esquecer.

Por fim, agradeço à CAPES por incentivar esta pesquisa e, além de tudo, incentivar a minha vida acadêmica.

RESUMO

Criar e avaliar jogos manualmente é uma tarefa árdua e trabalhosa. Para automatizar este processo, existem soluções que abrangem jogos de tabuleiro, como xadrez, damas e Go. Porém, essas soluções não englobam jogos de mesa, que possuem dados, cartas e mapas. Este trabalho propõe uma abordagem de design evolutivo de jogos de mesa, avaliando o processo com a geração de variantes do jogo Risk. Os resultados mostram novas versões, com mapas menores, resultando em partidas mais curtas. Além disso, as variantes produzem partidas mais equilibradas. Assim, o *pipeline* de design evolutivo proposto é capaz de criar e evoluir um jogo de mesa com diferentes equipamentos, além de gerar melhores versões do jogo Risk.

Palavras-chave: *Evolutionary Game Design*. Algoritmo Genético. Risk.

Evolutionary Tabletop Game Design: A Case Study in the Risk Game

ABSTRACT

Creating and evaluating games manually is an arduous and laborious task. To automate this process, there are solutions that cover board games such as chess, checkers and Go. However, these solutions do not include table games, which have dice, cards and maps. This work proposes an evolutionary design approach for tabletop games, evaluating the process by generating variants of the game Risk. The results show new versions, with smaller maps, resulting in shorter matches. Also, the variants produce more balanced matches, but don't leave out the usual drama. Thus, the proposed evolutionary design pipeline is capable of creating and evolving a tabletop game with different equipment, in addition to generating better versions of the Risk game.

Keywords: *Evolutionary Game Design. Genetic Algorithm. Risk.*

LISTA DE ABREVIATURAS E SIGLAS

GA	Genetic Algorithms
GDL	Game Description Language
GGP	General Game Playing
IA	Inteligência Artificial
MCTS	Monte Carlo Tree Search

LISTA DE FIGURAS

Figura 2.1	Mapa do jogo Risk base.....	16
Figura 2.2	Mapa do jogo War base.	17
Figura 2.3	CrITÉrios de avaliaÇo de um jogo.....	23
Figura 2.4	Fluxograma de um algoritmo genético.....	25
Figura 2.5	RepresentaÇo do <i>crossover</i> de 1 ponto e de 2 pontos.	27
Figura 2.6	RepresentaÇo do <i>crossover</i> uniforme.....	28
Figura 4.1	Pipeline implementado na geraÇo de verses do jogo Risk. Cada nova populaÇo preenchida representa uma geraÇo. A execuÇo termina quando o nmero de geraÇes estabelecido for atingido.	35
Figura 4.2	VisualizaÇo do mapa no inÍcio de uma partida.	37
Figura 4.3	VisualizaÇo do mapa aps a alocaÇo de tropas.....	38
Figura 5.1	<i>Fitness</i> resultante das melhores execuÇes.....	53
Figura 5.2	<i>Fitness</i> ao longo da execuÇo que resultou o melhor <i>fitness</i> e o segundo melhor <i>fitness</i> ao final do processo.	55
Figura 5.3	VariaÇo do <i>fitness</i> ao longo dos valores de geraÇo.	57
Figura 5.4	VariaÇo do <i>fitness</i> ao longo dos valores de tamanho da populaÇo.....	58
Figura 5.5	Valores de <i>fitness</i> ao longo dos valores de tamanho do torneio.....	58
Figura 5.6	Valores de <i>fitness</i> ao longo dos valores de taxa de mutaÇo.....	59
Figura 5.7	RepresentaÇo do mapa inicial 9.	60
Figura 5.8	RepresentaÇo do mapa do melhor indivÍduo da execuÇo rpida.	62
Figura 5.9	RepresentaÇo do mapa do melhor indivÍduo da execuÇo longa.	62
Figura A.1	Sequncia de comandos na fase de alocaÇo.	76
Figura A.2	Sequncia de comandos na fase de ataque, adicionando tropas.	76
Figura A.3	Sequncia de comandos na fase de ataque, adicionando bnus de continente.....	76
Figura A.4	Sequncia de comandos na troca de cartas.	77
Figura A.5	Sequncia de comandos na fase de ataque, atacando o oponente.....	77
Figura A.6	Sequncia de comandos na fase de fortificaÇo.....	78

LISTA DE TABELAS

Tabela 2.1	Relação de ganho de tropas com base na ordem de troca de cartas.	19
Tabela 2.2	Resumo das diferenças entre o Risk e o War.	21
Tabela 4.1	Valores possíveis de cada parâmetro.	40
Tabela 4.2	Valores ótimos para os critérios de avaliação.	50
Tabela 5.1	Parâmetros de cada execução geradora dos 10 melhores <i>fitness</i>	54
Tabela 5.2	<i>Fitness</i> encontrados em gerações não finais cujo valor é menor do que o menor valor encontrado anteriormente entre todas as gerações finais.	56
Tabela 5.3	Parâmetros das execuções mais simples e mais exaustiva.	61
Tabela 5.4	Critérios das execuções mais rápida e mais longa.	63
Tabela 5.5	Valores dos critérios calculados para os parâmetros originais de Risk e War.	65
Tabela 5.6	Número de indivíduos gerados com os parâmetros do War nas 10 me- lhores configurações.	67

SUMÁRIO

1 INTRODUÇÃO	11
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 Jogos	13
2.2 Risk, War e variações	15
2.2.1 Fluxo de jogo	17
2.2.1.1 Receber e posicionar tropas	18
2.2.1.2 Atacar oponentes	19
2.2.1.3 Fortificar territórios e receber carta	19
2.2.2 Detalhes e diferenças	20
2.3 Design evolutivo de jogos	21
2.4 Algoritmos Genéticos	24
2.4.1 Seleção e avaliação	26
2.4.2 Reprodução	26
3 TRABALHOS RELACIONADOS	29
3.1 Risk, War e suas aplicações	29
3.2 Geração automatizada de jogos	31
3.3 Conclusões	34
4 DESIGN EVOLUTIVO DE VARIANTES DE RISK	35
4.1 Risk simplificado	36
4.2 Parâmetros	37
4.3 Algoritmo Genético	40
4.3.1 População Inicial	40
4.3.2 Seleção	41
4.3.3 <i>Crossover</i> e Mutação	41
4.3.4 Testes do Jogo	42
4.3.4.1 Agente baseado em regras	43
4.3.5 Avaliação	44
4.3.5.1 Completude	45
4.3.5.2 Duração	45
4.3.5.3 Vantagem	46
4.3.5.4 Fator de ramificação	46
4.3.5.5 Drama	48
4.3.5.6 Movimentos extremos (<i>killer moves</i>)	48
4.3.5.7 Mudança de Liderança	49
4.3.5.8 <i>Fitness</i>	49
4.3.6 Substituição	50
4.4 Resumo	51
5 EXPERIMENTOS	52
5.1 Hiperparâmetros	52
5.2 Evolução do Mapa	59
5.3 Evolução dos Jogos	63
5.4 Jogabilidade, Versões Geradas e Versões Reais	65
5.5 De Risk a War	66
5.6 Resumo	68
5.7 Análise	68
6 CONCLUSÃO	70
REFERÊNCIAS	71
APÊNDICE A — INSTRUÇÕES PARA UM JOGADOR HUMANO	75

1 INTRODUÇÃO

Os jogos estão muito presentes nas nossas vidas, desde a infância até a fase adulta. Porém, os jogos vão muito além da diversão. Suas características como, complexidade computacional, variabilidade de estratégias e comportamentos emergentes atraem o interesse da pesquisa acadêmica (RISI; PREUSS, 2020).

Porém, criar e testar jogos é uma tarefa árdua e custosa. É preciso uma grande quantidade de partidas para decidir se um jogo é bom o suficiente para ser publicado. Nesse sentido, a Inteligência Artificial (IA) se torna uma grande aliada, podendo realizar estes testes de uma forma mais eficiente, tornando a procura por jogos balanceados e divertidos muito mais rápida.

Entre os diversos métodos existentes, a utilização de métodos evolutivos na avaliação e geração de novos jogos se mostrou efetiva com a introdução do sistema Ludi (BROWNE, 2011), que é capaz de gerar e avaliar automaticamente jogos de tabuleiro, como Xadrez e Damas, com o uso de *design* evolutivo de jogos. O Ludi foi capaz de criar um jogo completamente novo, chamado Yavalath, e que fez relativo sucesso entre os jogadores, como relatado por Browne (2011). Para isso, a implementação do *design* evolutivo aplica programação genética, evoluindo conjuntos de regras, e diversas formas de avaliação dos jogos criados. A grande dificuldade, como cita Browne (2011), é que um computador não consegue criar novas regras.

Por mais que um jogo comercial tenha sido criado, provando seu potencial, o Ludi só consegue trabalhar com jogos tradicionais de tabuleiro, possuindo somente as peças do jogo. Equipamentos como cartas, dados e mapas presentes em jogos de mesa não tem suporte na implementação da Ludi, por exemplo. Estes equipamentos são as principais diferenças entre os dois gêneros. As cartas e os dados adicionam uma complexidade a mais com a aleatoriedade e incerteza.

Este trabalho se propõe a estender *design* evolutivo de jogos para jogos de mesa e aplicá-lo em Risk, cuja variante brasileira é o War, um jogo de mesa onde os jogadores batalham para conquistar todos os territórios dispostos no mapa. Os ataques e defesas são decididos com base em lançamentos de dados, adicionando incerteza na partida. Além disso, cartas podem ser trocadas por tropas, aumentando as possibilidades estratégicas do jogo. Diversos trabalhos utilizam a problemática do Risk, tanto para criar um agente capaz de jogá-lo (WOLF, 2005; OLSSON, 2005), quanto para ensinar conceitos como geopolítica e política internacional (ARAÚJO et al., 2018; MARKS, 1998). Mas o jogo

é pouco explorado em termos de *design* e IA.

O objetivo do trabalho é utilizar *design* evolutivo de jogos na criação e avaliação de novas versões do jogo Risk. Uma primeira investigação de *design* evolutivo de jogos em Risk foi feita por Bombardelli (2022) usando programação genética como algoritmo evolutivo. O presente trabalho amplia o escopo de investigação, adicionando elementos do jogo original que estavam ausentes e substituindo a programação genética por um algoritmo genético, garantindo maior exploração do espaço de busca.

Ao fim dos experimentos, o *pipeline* proposto foi capaz de gerar variantes do Risk com mapas menores, resultando em partidas mais curtas. Porém, uma limitação foi a predominância de mapas muito pequenos que, apesar de terem uma boa avaliação na função objetivo, geram partidas triviais.

As principais contribuições deste trabalho são: um aprofundamento da investigação de *design* evolutivo de jogos de mesa, a descoberta de que tentar minimizar o número de movimentos possíveis (fator de ramificação) favorece a predominância de mapas pequenos e disponibilização do código-fonte de toda a implementação¹ para fins de reprodutibilidade. Este trabalho pode ser um ponto de partida para expandir o estudo de *design* evolutivo de jogos para Risk e outros jogos de mesa, além de mostrar que o Risk pode ser largamente explorado no âmbito de IA.

O restante deste trabalho é estruturado da seguinte forma. O Capítulo 2 fornece o conhecimento básico para o entendimento do trabalho, como definições de jogo, das regras de Risk e War e do *design* evolutivo de jogos. O Capítulo 3 apresenta os trabalhos relacionados com os temas abordados, expondo o estado da arte. O Capítulo 4 discorre sobre o *pipeline* desenvolvido, mostrando as etapas necessárias nas gerações de novas versões de Risk. O Capítulo 5 relata os experimentos realizados, assim como os resultados obtidos através dele. Por fim, o Capítulo 6 conclui tudo o que foi apresentado, falando também sobre trabalhos futuros.

¹<https://github.com/lanabr/Risk-Generation>

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, serão apresentados os principais conceitos utilizados na elaboração desta dissertação. Na Seção 2.1, serão apresentadas a definição de jogos e suas principais características. Na Seção 2.2, serão expostos os jogos de mesa Risk e War, suas regras e especificações. Estes foram largamente explorados no processo de design evolutivo de jogos, que será evidenciado na Seção 2.3.

2.1 Jogos

Existem várias formas de definir jogos em diversos aspectos diferentes. Entre elas, a definição de Kelley (2013, p. 40) dá destaque para a existência de regras, dizendo que "*[A] game is a form of recreation constituted by a set of rules that specify an object to be attained and the permissible means of attaining it*". Dessa forma, Kelley (2013) continua sua explicação dizendo que justamente as regras são o que diferenciam os jogos de outras formas de recreação, como danças, viajar e até mesmo outros passatempos.

O envolvimento de jogadores pode ou não ser importante na hora de definir um jogo. Um jogo pode existir sem a presença de um jogador, em sua forma conceitual, com seus materiais e regras. Porém, um jogador não existe sem a presença de um jogo do qual ele é jogador (AARSETH, 2014). Essa constatação liga o jogador a um jogo específico, ou seja, um jogador em potencial só se torna de fato um jogador real quando toma conhecimento das instruções de um determinado jogo, assim como quando o joga pela primeira vez.

A possibilidade de ganhar ou perder faz dos jogos um grande cenário para a competição, seja jogando sozinho, contra um oponente ou em grupo. Além disso, mesmo tendo a diversão como principal atrativo, o aprendizado se torna um efeito interessante a longo prazo. Segundo Dempsey,

[...] a game involves some aspect of competition, even if that competition is with oneself. Most games are intended to be entertaining, not instructional. Often, the reason a person chooses to play a game is to experience the fun of engaging in the gaming activity. Learning is usually incidental or intentional only for the purposes of one becoming a better gamer. (DEMPSEY et al., 2002, p. 159)

Porém, após analisar diferentes formas de descrever e caracterizar jogos, Salen and Zimmerman (2003) moldaram uma nova definição, cortando conceitos dispensáveis e destacando elementos essenciais. Com isso, Salen e Zimmerman utilizam a seguinte

definição:

A game is a system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome. (SALEN; ZIMMERMAN, 2003, p 81)

Dessa forma, Salen e Zimmerman sintetizam a definição de um jogo e encontrar as peças-chave para um modelo básico. As ideias principais dessa definição são apresentadas abaixo.

- Sistema: É a conexão entre todos os elementos envolvidos em um jogo. Separados, cartas, dados e tabuleiros não fazem muito sentido. Porém, quando juntos, ganham relações específicas, fazendo-os ter sentido nesse meio.
- Jogadores: Um jogo existe para ser jogado por um ou mais jogadores. Esses jogadores interagem entre si e com o jogo para experienciar suas possíveis jogadas.
- Artificialidade: O espaço em que uma partida está acontecendo é, claramente, no mundo real. Porém, todo o seu contexto acontece artificialmente, suas regras são inventadas, juntamente com os materiais utilizados.
- Conflito: Geralmente é a peça central do jogo, sempre há um conflito a ser resolvido, seja com o sistema ou entre os jogadores. Há uma disputa de poder.
- Regras: Ditam a estrutura do jogo, como ele deve ser jogado, delimitando até onde os jogadores podem ir, o que podem e o que não podem fazer.
- Resultado quantificável: Quando uma partida termina, deve ser possível dizer se o jogador perdeu ou ganhou, ou se ganhou uma quantidade de pontos.

Condensando estes conceitos, Browne (2011) utilizou o modelo *means-plays-ends* em seu trabalho, tanto na descrição dos jogos, na criação deles e na sua avaliação. Uma tradução livre desse modelo é feita abaixo.

- Recursos (em inglês, *means*): são os equipamentos e as regras necessárias para jogar.
- Jogabilidade (em inglês, *plays*): é a interação entre os jogadores, definida pelas estratégias pensadas e movimentos feitos a partir disso.
- Consequências (em inglês, *ends*): são os resultados gerados a partir destes movimentos.

Estas definições apresentadas acima são importantes para delimitar quais tipos de jogos foram utilizados no trabalho de Browne (2011) e que também serão vistos nessa dissertação. Nesse sentido, dentre os muitos jogos que podem se encaixar nesse modelo,

o presente trabalho utiliza jogos com características parecidas com as dos jogos combinatoriais. Os jogos presentes neste trabalho possuem as seguintes características:

- Finitos: Produzem resultados bem definidos.
- Discretos: É jogado em turno, onde cada jogador joga em um turno diferente.
- Não determinísticos: Dependem de sorte ou outros elementos aleatórios.
- Informação perfeita: Todos os jogadores têm visão total do estado do jogo.
- Mais de dois jogadores: mais de dois jogadores podem estar presentes em uma partida, trazendo características mais elaboradas.

Dessas características apresentadas, os jogos combinatoriais se diferem em duas. Eles são determinísticos, não dependendo da aleatoriedade, e possuem somente dois jogadores. Exemplos desses tipos de jogos são xadrez, damas e jogo da velha.

Jogos com estas características são excelentes casos de teste, tendo uma profundidade razoável, mas também sendo conduzidos por regras simples e com definições bem delimitadas. Este trabalho, contudo, não foca em explicar ou aplicar a Teoria dos Jogos Combinatoriais em nenhum viés, somente em utilizar jogos com as características definidas por esta teoria.

2.2 Risk, War e variações

Os jogos Risk e War são jogos de mesa onde os jogadores batalham entre si para conquistar todos os territórios presentes no mapa. Isso é feito por meio de ataques utilizando peças, chamadas de tropas, e dados.

Fazendo uma pequena retrospectiva, no ano de 1957, Albert Lamorisse, um famoso cineasta francês, inventa o jogo "La Conquête du Monde", (A Conquista do Mundo, em português)¹. Porém, ele só existia na França. Em 1959, os irmãos Parker, donos da empresa de jogos Hasbro, decidiram publicar o jogo nos Estados Unidos. Fizeram algumas modificações nas regras para torná-lo mais atrativo para iniciantes, assim como melhorias no seu design, e lançaram o jogo com um novo nome, Risk: *The Continental Game*. No Brasil, motivados pela falta de jogos de tabuleiro no país, quatro amigos criaram a empresa Grow, em 1972. Seu primeiro jogo lançado foi o War, baseado no Risk

¹<https://www.ultraboardgames.com/risk/history.php>

dos irmãos Parker². Ao longo dos anos, os dois jogos, Risk³ e War⁴, lançaram ainda mais variações, com mudanças nos mapas e na duração das partidas. A versão clássica do mapa do Risk pode ser visto na Figura 2.1 e o mapa do War, na Figura 2.2.

Figura 2.1 – Mapa do jogo Risk base.



Fonte: <https://gamenightblog.com/2019/05/16/risk-family-game-review/>

Estes dois jogos tem princípios parecidos. Suas semelhanças e diferenças são peculiares, mas podem mudar toda a dinâmica, o tempo de jogo e, inclusive, podem alcançar mais entusiastas como consequência destas pequenas mudanças.

Além do tabuleiro e das peças que representam as tropas, os dois jogos possuem cartas que representam todos os territórios disponíveis no mapa. Em específico no jogo War, essas cartas possuem um propósito a mais que no Risk. Elas têm como objetivo distribuir os territórios entre os jogadores no início de cada partida. Além de conter os respectivos territórios, cada carta contém também um símbolo, podendo esse variar conforme a versão do jogo. Estes símbolos são utilizados na troca de cartas por tropas, sendo esse um dos modos de obter tropas.

Existe também outro tipo de carta que se faz presente em ambos os jogos. Ao contrário das cartas de território, as cartas de objetivo são utilizadas para dar ao jogador um objetivo sobre o qual ele fundamentará sua estratégia de jogo. Porém, mesmo estando disponível no Risk, as cartas de objetivo são indispensáveis somente no War, conforme os

²<https://vejasp.abril.com.br/coluna/memoria/os-45-anos-de-um-jogo-classico-o-war-da-grow/>

³<https://www.hasbro.com/common/instruct/risk.pdf>

⁴https://tablegames.com.br/wp-content/uploads/2017/10/war_manual_table_games.pdf

Figura 2.2 – Mapa do jogo War base.



Fonte: <https://www.esfera.com.vc/p/jogo-war-tabuleiro-o-jogo-da-estrategia-war-edicao-especial-grow/e100011445>

manuais de cada jogo.

Além disso, geralmente existem 3 tipos de tropas nestes jogos. No Risk original, existe a Infantaria, que representa 1 tropa, a Cavalaria, representando 5 tropas, e a Artilharia, que representa 10 tropas. Essas interpretações são motivadas principalmente pelo limite de espaço de cada território no tabuleiro, além de reduzir a quantidade de peças em cada edição física do jogo.

Nas próximas subseções, as regras dos jogos War e Risk serão apresentadas com mais detalhes, bem como as alterações que resultam em suas diferenças. Boa parte da compatibilidade se dá na sequência das etapas do jogo e nas regras de uma forma geral, enquanto as disparidades ocorrem em pontos-chave dessas etapas.

2.2.1 Fluxo de jogo

O fluxo da partida é o que mais torna os dois jogos similares. No início da partida, cada jogador começa com uma determinada quantidade de territórios referente à divisão do número total pela quantidade de jogadores. Após isso, começa a fase de turnos, que consiste em quatro etapas:

- Receber e posicionar tropas;
- Atacar oponentes;
- Fortificar territórios.
- Receber carta de território (se conquistou pelo menos um território)

As particularidades de cada etapa serão expostas a seguir. Distribuir as tropas recebidas e fortificar os territórios atuais são etapas obrigatórias. Já atacar os oponentes e receber uma carta são fases que podem ser omitidas.

2.2.1.1 Receber e posicionar tropas

No início de cada turno, os jogadores recebem tropas de diversas formas diferentes. Essas tropas devem ser posicionadas nos territórios que o jogador possui de acordo com regras específicas. Existem quatro modos de adquirir tropas:

- *Territórios ocupados*: o jogador recebe novas tropas baseado na quantidade de territórios que ele já possui. Essa conta é feita somando o total de territórios ocupados e dividindo por um número que depende das regras da versão jogada.
- *Continentes inteiros ocupados*: para cada continente inteiramente ocupado pelo jogador, ele recebe a quantidade de tropas relativa àquele continente. Essa relação fica impressa no tabuleiro para ser consultada quando necessário. A quantidade de tropas ganhas por continente varia conforme a quantidade de territórios que aquele continente possui. Da mesma forma, as tropas ganhas como consequência de possuir um continente devem ser distribuídas somente no mesmo continente.
- *Troca de cartas*: o jogador tem a possibilidade de trocar as cartas conquistadas em turnos passados. Três cartas podem ser trocadas por vez e, quando o jogador alcançar cinco cartas, ele tem a obrigação de realizar uma troca. Cada troca concede ao jogador o número de tropas relativo à quantidade de trocas já feitas em toda a partida até aquele momento. Conforme mostrado na Tabela 2.1, a 1ª troca transfere 4 tropas ao jogador, a 2ª dá 6 tropas, e assim por diante. A partir da 6ª troca, o número de tropas concedidas ao jogador aumenta de 5 em 5 unidades. Essa quantidade de tropas ganhas pelo jogador pode ser distribuída em qualquer território desejado, segundo a estratégia definida.
- *Território ocupado nas cartas trocadas*: considera novamente as cartas trocadas anteriormente. Se dentre alguma dessas cartas estiver um território ocupado pelo

Tabela 2.1 – Relação de ganho de tropas com base na ordem de troca de cartas.

Troca	1 ^a	2 ^a	3 ^a	4 ^a	5 ^a	6 ^a	7 ^a	8 ^a	9 ^a	10 ^a	11 ^a	12 ^a	13 ^a	14 ^a
Nº de Tropas	4	6	8	10	12	15	20	25	30	35	40	45	50	55

Fonte: A Autora

jogador, ele ganha duas tropas adicionais que devem ser colocadas exclusivamente no território em questão.

2.2.1.2 Atacar oponentes

Quando estiver em seu turno, o jogador pode atacar somente territórios inimigos que façam fronteira com os seus territórios. Se desejar investir, o jogador atacante deve escolher com quantas tropas deseja atacar, e o mesmo acontece com o defensor. Porém, mesmo se possuir mais tropas, o máximo de dados disponíveis para utilizar é limitado a 3 para o ataque e pode variar entre 2 e 3 para a defesa, dependendo da versão do jogo. Ainda, o jogador atacante deve possuir uma unidade de tropa a mais do que a quantidade de dados que deseja utilizar. Esta tropa remanescente serve para marcar a posse do território se o jogador acabar perdendo todas as tropas apostadas na rolagem de dados.

A cada ataque realizado, os jogadores envolvidos jogam a quantidade de dados desejada e comparam os resultados em ordem decrescente. A cada comparação, o jogador que perdeu deve remover uma tropa de seu território e, em caso de empate, o ganhador é o que está defendendo. Por exemplo, o jogador atacante rolou os números 5, 4 e 1 e o defensor rolou os dados 6, 3 e 1. No primeiro confronto, o defensor ganha e o atacante deve retirar uma tropa de seu território. No segundo confronto, o atacante ganha e o defensor deve subtrair uma tropa. Por fim, há um empate no terceiro confronto, fazendo o atacante perder mais uma tropa.

Ao conquistar um território, o jogador deve mover a quantidade de tropas desejada para sua nova aquisição. O número de tropas a ser deslocado também varia de acordo com a versão do jogo. A fase de ataque pode terminar a qualquer momento, dependendo somente da estratégia dos jogadores.

2.2.1.3 Fortificar territórios e receber carta

Após o ataque, o jogador entra na fase de fortificação. Ele pode então mover suas tropas entre os territórios como preferir. Porém, cada tropa só pode ser movida uma vez,

não podendo seguir por mais territórios do que os da fronteira do território original.

No final do seu turno, se o jogador conquistou pelo menos um território, ele recebe uma carta. Essa carta poderá ser trocada no início de cada turno subsequente, conforme mostrado anteriormente na etapa de receber e posicionar tropas.

Ao longo da introdução das regras dos jogos, foi possível observar que o fluxo de jogo tanto do Risk quanto do War é muito próximo, poucos detalhes ficam ausentes.

2.2.2 Detalhes e diferenças

A primeira grande diferença está no objetivo de jogo. No Risk, no modo básico, o objetivo é conquistar todos os territórios do mapa. Existe uma versão em que as cartas de objetivo são utilizadas para ditar um rumo e estratégias diferentes para cada jogador. Porém, conforme o livro de regras, essa é uma *variação* do jogo base. Já no War, o modo com as cartas de objetivo já é o *principal*.

A próxima diferença surge logo no começo da partida. Quando os jogadores precisam ocupar seus territórios iniciais: no Risk, os jogadores alternam a vez e *escolhem* os territórios de acordo com a sua estratégia pessoal, de forma livre. Já no War, as cartas de território são distribuídas de forma aleatória entre os jogadores e cada um *ocupa os territórios designados a ele*.

Outra diferença é no cálculo de tropas recebidas a cada início de turno. No Risk, a quantidade de tropas a ser recebida é a soma da quantidade de territórios dividido por três. Já no War, a quantidade de territórios é dividida por dois, aumentando a quantidade de tropas a serem colocadas, e, por consequência, aumentando o tempo de jogo. Isso acontece porque, quanto mais tropas posicionadas nos territórios, maiores são as chances do jogador conseguir defender o seu território.

A próxima diferença está presente na fase de ataque do jogo. Enquanto no Risk o jogador pode rolar no máximo dois dados para se defender, no War ele consegue utilizar até 3 dados. Essa pequena diferença pode alongar ainda mais uma partida jogada conforme as regras do War, uma vez que o jogador atacante pode perder até 3 tropas e diminuir, assim, a quantidade de ataques disponíveis naquele turno.

Durante o ataque, outra diferença é importante. Quando um território é adquirido, uma quantidade de tropas deve passar do território atacante para o atacado. No Risk, essa quantidade pode variar desde o número de dados jogados até todas as tropas do território, menos uma que deve ficar para demarcar a posse. Já no War, pode variar entre uma tropa

até a quantidade de dados jogados. Seguindo as regras do Risk, fica mais fácil fazer ataques em sequência e abrir um caminho entre territórios inimigos, porém os territórios atacados e conquistados podem ficar desprotegidos com poucas tropas. Já pelas regras do War, o jogador fica mais preso a ataques locais, sem muita possibilidade de expansão, porém todos os seus territórios ficam seguros com poucas tropas subtraídas a cada ataque.

Estas diferenças entre dois jogos tão parecidos foram um alerta para como a mudança de detalhes pode resultar em variações com qualidades. Na Tabela 2.2, é apresentado um resumo das diferenças entre os jogos Risk e War.

Tabela 2.2 – Resumo das diferenças entre o Risk e o War.

Tópico	Risk	War
Objetivo do jogo	Conquistar todos os territórios	Alcançar o objetivo da carta escolhida
Ocupação de territórios	Livre escolha	Designação por cartas
Fator do cálculo de tropas	3	2
Dados da defesa	2	3
Tropas movidas na conquista de território	Número de dados usados até o total de tropas do território menos 1	Uma tropa até a quantidade de dados utilizados

Fonte: A Autora

2.3 Design evolutivo de jogos

Criar jogos divertidos e originais não é uma tarefa fácil. Com o objetivo de gerar novos jogos com conjuntos de regras interessantes e de qualidade, Browne (2011) propôs o *framework* Ludí. Além de gerar jogos, o sistema também joga e avalia as novidades criadas. Os principais componentes deste *pipeline* são:

1. Linguagem de Descrição de Jogos (GDL, do inglês *Game Description Language*): Define as regras do jogo e seus componentes.
2. Jogador Geral de Jogos (GGP, do inglês *General Game Playing*): É capaz de interpretar e jogar qualquer jogo descrito na GDL.

3. Módulo de Estratégia: Comunica o planejamento das jogadas.
4. Módulo de Avaliação: Mede a qualidade dos jogos.
5. Módulo de Síntese: Gera novos jogos.

No geral, uma GDL tem a missão de ser robusta e abrangente, a fim de representar jogos conhecidos e desconhecidos, e também precisa ser compreensiva e coerente para os criadores humanos. Algumas GDLs conhecidas são a ZRF, utilizada pela *Zillions of Games*⁵, e a Stanford GDL, utilizada nas competições AAAI GGP (GENESERETH; LOVE; PELL, 2005).

A GDL da Ludi é baseada no sistema de *ludemes*, cunhado pela primeira vez por Borvo and Seguin (1977), unidades de informação de um jogo que podem ser replicadas e transformadas, muitas vezes sendo compatíveis com as regras e equipamentos. Os *ludemes* podem compreender uma informação unitária, como abaixo, onde os *ludemes* descrevem características de um tabuleiro:

```
(tiling square)
(size 3 3)
```

A primeira linha declara que o tabuleiro será composto de unidades quadradas. Já a segunda linha expõe que o tabuleiro será de tamanho 3x3. Também, os *ludemes* podem encapsular um conjunto de outros *ludemes* correlacionados. Abaixo, o *ludeme* descreve o tabuleiro em si, e não mais informações separadas como antes:

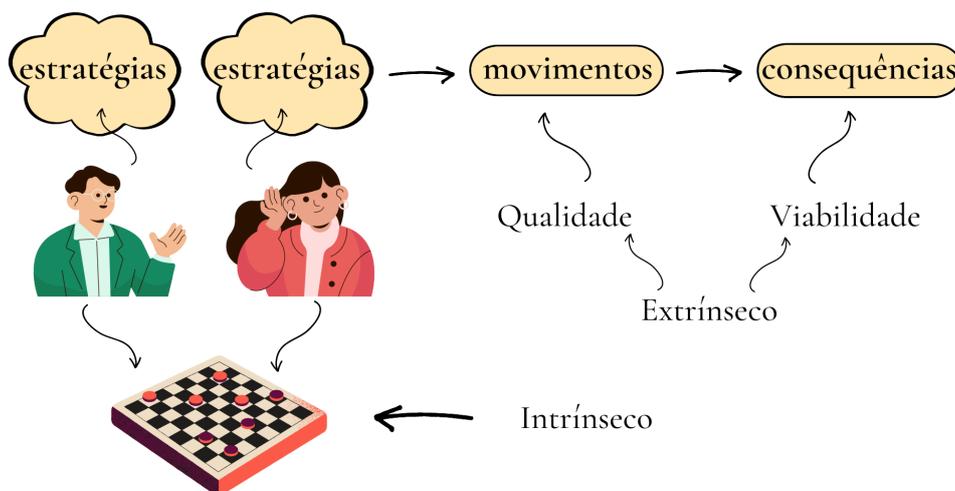
```
(board
  (tiling square)
  (size 3 3)
)
```

Conjuntos de *ludemes* descrevem as regras de um jogo, podendo estar aninhados em praticamente qualquer nível. Além disso, descrevem os equipamentos e as recompensas ganhas ao final da partida. Com essa representação simples é possível descrever uma grande variedade de jogos de forma padronizada, fazendo com que todos os jogos sejam compreendidos da mesma forma.

Um GGP (PITRAT, 1968), de forma geral, é um jogador capaz de jogar qualquer jogo descrito na GDL em que é especializado. Ele consegue jogar bem, mas não é especialista em nenhum jogo. O GGP do sistema da Ludi é responsável por analisar o arquivo contendo as regras do jogo na GDL da Ludi e identificar se existe algum erro gramatical. Além disso, ele administra o planejamento de jogadas e momentos em que todos os

⁵<https://www.zillions-of-games.com/>

Figura 2.3 – Critérios de avaliação de um jogo.



Fonte: Adaptado de Browne (2011)

jogadores passam a vez, por exemplo.

O Módulo de Estratégia é responsável por encontrar o movimento a ser feito em determinada rodada. Isso é feito com base no sistema de conselheiros e políticas. Os conselheiros são características visíveis do estado do jogo naquele momento, dando uma noção de quão bem o jogador está com base no seu objetivo. As políticas são os pesos de cada conselheiro, ou seja, o quão significativa é aquela informação para o tipo do jogo. Por exemplo, uma política razoável para o jogo da velha deve desconsiderar o conselheiro de captura, já que a captura não faz parte das regras.

O Módulo de Avaliação realiza a medição dos critérios de avaliação, trazendo assim um parâmetro de classificação. São medidas características intrínsecas e extrínsecas dos jogos. Os critérios intrínsecos são medidos através das regras e dos equipamentos, sem realizar qualquer teste. Já os extrínsecos são divididos em critérios de viabilidade e de qualidade, e são calculados com base em métricas retiradas ao longo de diversas partidas jogadas, seja durante ou ao final da partida.

Os critérios de qualidade são mais difíceis de calcular e expressar em números. Eles explicam a jogabilidade e a interação dos jogadores com as regras e os equipamentos. Os critérios de viabilidade revelam características das consequências dessas ações, sendo então a avaliação fundamental para identificar jogos sem futuro. Esses critérios foram apresentados na Figura 2.3.

Ao todo, são medidos 16 critérios intrínsecos, 11 critérios de viabilidade e 30 de qualidade. Utilizando todas essas medidas, (BROWNE, 2011) realizou um teste, classifi-

cando jogos automática e manualmente, com a ajuda de voluntários. Com isso, conseguiu encontrar os critérios mais relevantes, utilizando os 16 mais significativos na fase de avaliação no sistema da Ludi.

Por fim, no Módulo de Síntese, novos conjuntos de regras são criados por meio de algoritmos evolutivos. O conjunto de indivíduos passa por seleção, *crossover*, mutação e segue por uma série de validações. São verificados um conjunto de exigências quanto a regularidade das regras criadas, além da escolha de um nome para o jogo.

No sistema da Ludi, são utilizados como população inicial jogos reais com diversidade de regras. Durante o processo, a variabilidade genética é mais valorizada do que perfeição. Regras não utilizadas são mantidas, considerando a possibilidade de uso futuro. Ao final do *pipeline*, é feita uma avaliação dos jogos criados naquela iteração, utilizando o módulo apresentado anteriormente.

A escolha da programação genética para este tipo de problema automatiza a responsabilidade pela criação do jogo criado. Além disso, o próprio sistema deve conseguir reconhecer quais regras são mais importantes e quais não fazem tanta diferença. A aparição de regras inesperadas também motiva a utilização deste método.

A versão atual é o Ludii⁶, segunda versão do Ludi. Apesar de ter consideráveis evoluções, este sistema ainda não realiza a geração de novos jogos, apenas possuindo a linguagem de descrição e o agente.

2.4 Algoritmos Genéticos

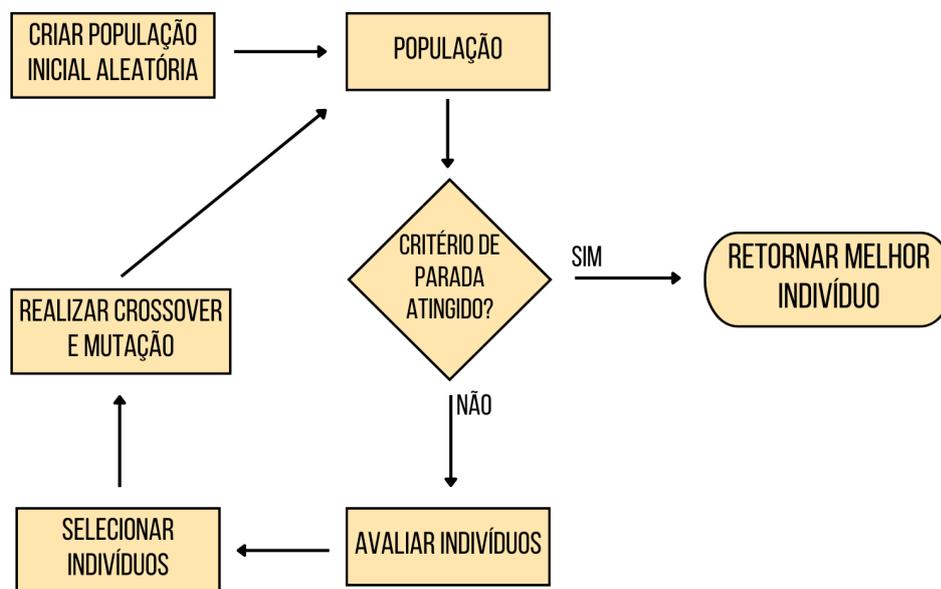
A primeira menção a Algoritmos Genéticos (GA, do inglês *Genetic Algorithms*) foi feita em meados de 1960 por John H. Holland, segundo Mitchell (1995). Alguns anos depois, em 1975, os estudos foram formalizados pelo autor em conjunto com alunos e colegas (HOLLAND, 1975). Desde então, muitos avanços aconteceram, porém, a essência permanece a mesma.

Algoritmos genéticos, assim como outros métodos evolutivos, são baseados na teoria da evolução natural de Charles Darwin (DARWIN, 1859), na qual os indivíduos mais aptos sobrevivem e passam seus genes adiante. A ideia de adaptar estes conceitos de evolução para o meio computacional de tornou muito atrativa.

Computacionalmente, um GA é uma técnica que utiliza a metáfora da genética como método para resolver problemas de busca e otimização (SIVANANDAM; DEEPA,

⁶<https://ludii.games/>

Figura 2.4 – Fluxograma de um algoritmo genético.



Fonte: Adaptado de Sivanandam and Deepa (2008).

2008). A sequência básica de um GA pode ser visto na Figura 2.4.

Cada indivíduo codifica uma solução candidata para o problema de otimização, como uma sequência de argumentos de uma função, por exemplo. Geralmente, a população inicial é gerada de forma aleatória de acordo com os parâmetros selecionados para passarem pela evolução. A seleção destes parâmetros tem grande importância no processo evolutivo, sendo determinantes para o sucesso da busca da solução. Esta população inicial idealmente deve ter grande variedade de genes, ou seja, grande variedade nos valores dos parâmetros da função a ser otimizada, para assim melhorar as chances de encontrar uma boa solução.

A iteração de um algoritmo evolutivo básico consiste nas etapas de seleção, reprodução, avaliação e substituição. O algoritmo termina quando encontrar, entre a população, um indivíduo com uma avaliação ótima ou o número máximo de iterações ser atingido. Ao final de cada geração, os indivíduos gerados substituem a população atual e, conseqüentemente, serão os responsáveis pela criação da próxima geração.

2.4.1 Seleção e avaliação

Todos os indivíduos devem ser avaliados quanto a sua aproximação com relação à solução ótima. O cálculo do *fitness* pode ser feito de diversas formas, mas sempre representa o quão bom determinado indivíduo é. Quando há mais de um critério para determinar o quão perto um indivíduo está de ser a solução ótima, o impasse está em decidir qual critério é mais importante e o que fazer quando um indivíduo tem um valor bom para um critério e ruim para outro. Por isso, as decisões que envolvem o valor do *fitness* devem ser escolhidas com muito cuidado.

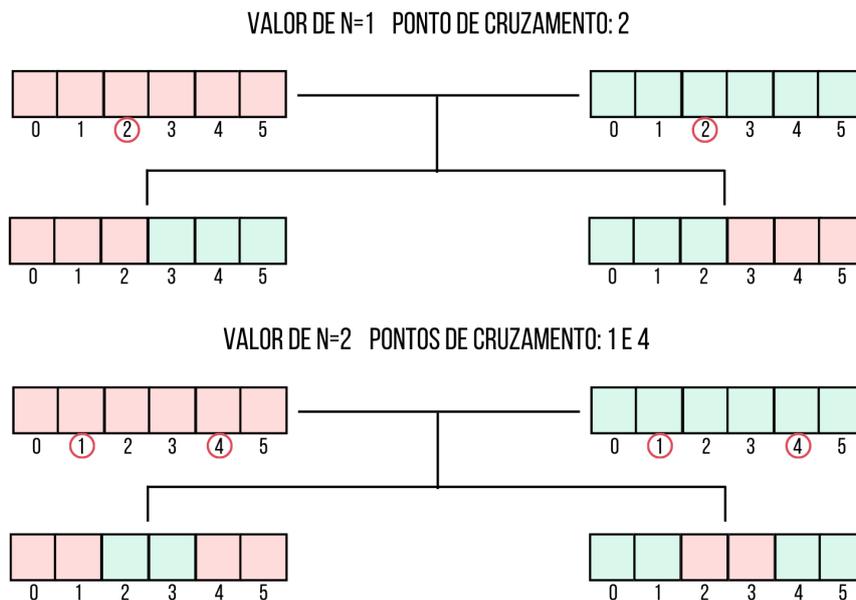
Durante a seleção, indivíduos são selecionados para transmitir os seus genes para a prole. Para isso, é preciso comparar os indivíduos entre si e selecionar os mais aptos, a fim de passar somente as melhores características para as próximas gerações. Porém, selecionar sempre os mesmos melhores indivíduos pode gerar uma prole sem diversidade. É preciso balancear as escolhas para ter mais chances de gerar o melhor indivíduo. Para definir quais indivíduos são os melhores, é preciso utilizar o valor do *fitness* apresentado anteriormente.

A seleção pode ser feita de diversas maneiras. No método da roleta, cada indivíduo recebe uma porção referente a probabilidade associada ao seu *fitness*. Assim, quando a roleta for "rodada", o melhor indivíduo tem a maior probabilidade de ser escolhido, e assim por diante. Desse modo, há uma probabilidade, por menor que seja, de um indivíduo médio ser escolhido e variar as características da prole. Porém, a grande questão é que o método da roleta favorece que o melhor indivíduo se replique muito rápido.

No método do torneio, k indivíduos aleatórios são selecionados para uma "disputa" e o de maior *fitness* vence e é selecionado. O melhor indivíduo daquela população pode não ser selecionado para o grupo, mas o melhor do grupo sempre será selecionado. O problema existente no método da roleta é amenizado no torneio, principalmente para valores pequenos de k . Existem outros métodos, porém todos se baseiam na aleatoriedade, como na roleta, ou na ordem, como no torneio.

2.4.2 Reprodução

A reprodução consiste na etapa de *crossover* e mutação. Nesta fase, os dois indivíduos selecionados gerarão dois novos indivíduos, a sua prole. Para isso, os parâmetros dos dois indivíduos escolhidos anteriormente, os pais, devem ser misturados. A etapa de

Figura 2.5 – Representação do *crossover* de 1 ponto e de 2 pontos.

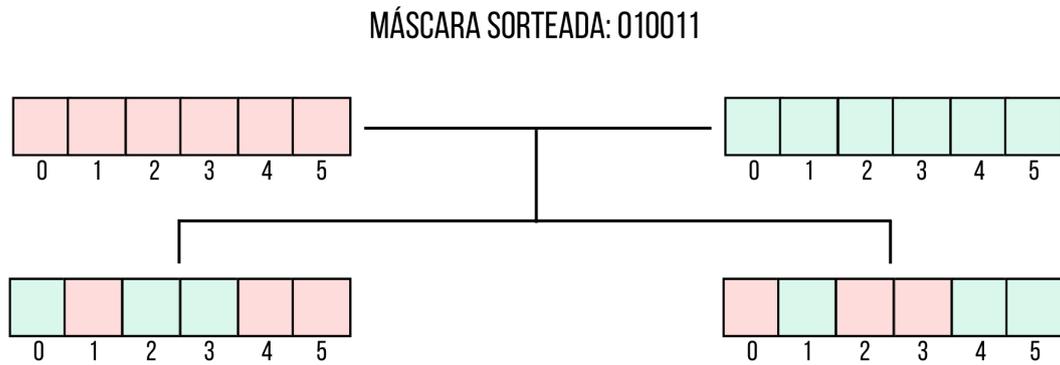
Fonte: A Autora.

crossover realiza essa tarefa, recombinando os pais e criando novos indivíduos. A etapa de mutação tem como objetivo diferenciar ainda mais a prole dos seus respectivos pais, mudando um valor para uma característica e assim prevenindo o algoritmo de ficar preso em mínimos locais. Isso permite maior exploração do espaço de soluções. Se o novo indivíduo tiver bom *fitness*, provavelmente seguirá nas próximas gerações.

Essa divisão de material genético que acontece no *crossover* pode ser feita de diversas maneiras. O *crossover* de n-pontos é o mais comum. As características dos dois indivíduos pais são divididas em n-pontos e fundidas alternadamente para formar os dois novos filhos. Assim, cada filho tem uma parte dos genes dos dois pais, sendo que um filho pode conter mais material de um pai do que de outro. Na Figura 2.5, são representados um exemplo de *crossover* de 1 ponto e de 2 pontos, onde cada filho recebe uma parte relativa a cada pai.

Já no *crossover* uniforme, uma máscara binária é sorteada para decidir de onde cada parâmetro virá. Se o valor do parâmetro na máscara for 1, o gene é herdado do primeiro pai, e se for 0, é herdado do segundo pai, e o contrário ocorre para o segundo filho, como mostrado na Figura 2.6.

Na mutação, os processos dependem também dos parâmetros. Uma ou mais características do indivíduo recebem novos valores de acordo com uma probabilidade, podem ser escolhidos aleatoriamente entre um conjunto de valores possíveis. Além disso, uma

Figura 2.6 – Representação do *crossover* uniforme.

Fonte: A Autora.

probabilidade de mutação deve ser adicionada, definindo a ocorrência da mutação. Como na natureza nem sempre uma mutação ocorre, o algoritmo também deve captar essa característica.

3 TRABALHOS RELACIONADOS

A geração automática de conteúdo para jogos já foi muito explorada (HENDRIKX et al., 2013; TOGELIUS et al., 2011), indo desde a criação de mapas e cartas até sons e texturas. Tanto jogos físicos quanto digitais já foram alvo de diversos pesquisadores que procuravam automatizar a criação de recursos (SUMMERVILLE; MATEAS, 2016; URIARTE; ONTAÑÓN, 2013). Porém, a geração de um jogo inteiro se torna mais complicada quando pensamos em todas as características importantes, como a coerência entre as regras e o quão divertido e interessante o jogo deve ser.

Nas próximas seções, serão explorados alguns trabalhos que utilizam Risk e War como plataforma de teste, além de pesquisas que tem como objetivo a criação de jogos por meio de métodos evolutivos. Estes trabalhos mostram uma tentativa de colocar estes dois temas em alta e apontar que ainda há muito a ser pesquisado e desenvolvido.

3.1 Risk, War e suas aplicações

Focando especialmente em Risk e War, grande parte das pesquisas foca em desenvolver agentes para jogar Risk de forma satisfatória. Diversos métodos são utilizados, como Busca de Monte Carlo em Árvore (MCTS, do inglês *Monte Carlo Tree Search*) (BRAND; KROON, 2014), além de uma abordagem multi-agentes, onde as decisões são tomadas por heurística (OLSSON, 2005), e Redes Neurais Convolucionais (CARR, 2020). Da mesma forma, outro trabalho utiliza uma combinação de MCTS e Redes Neurais (HEREDIA; CAZENAVE, 2021) e, por fim, há um trabalho que utiliza uma função de avaliação linear (WOLF, 2005).

Todas estas metodologias tem seus defeitos e qualidades, mas têm como propósito pavimentar o caminho de ambos os jogos e testar os diversos métodos existentes. Nos casos das abordagens baseadas em árvore, o ponto fraco recai sobre a função de avaliação. Por outro lado, quando a decisão da jogada é feita em pequenos passos, ou seja, decompondo a situação na análise dos países e não mais observando o mapa inteiro, a solução tende a ser simplificada e, com isso, melhor ponderada.

Os três primeiros trabalhos supracitados (BRAND; KROON, 2014; OLSSON, 2005; CARR, 2020) têm em comum a utilização do Lux Delux¹, uma plataforma online baseada em Risk, com IAs de diferentes perfis de jogo. É possível carregar mapas per-

¹<https://sillysoft.net/lux/>

sonalizados e suas próprias IAs, facilitando a etapa de testes. As principais desvantagens desta plataforma é a alteração de algumas regras do Risk original, como o valor da troca de cartas de território, e o fato de ser paga.

Outras pesquisas, contudo, focam em analisar situações mais específicas dos jogos. Harju (2012) analisa a probabilidade do jogador atacante conquistar um novo território, tendo como base as estatísticas de rolagem de dados. Diversos contextos devem ser considerados, como ataques em sequência, quando o defensor tem mais de 2 ou 3 tropas em seu território, ou ataques originados de mais de um território focando em um território específico. A interpretação sugerida pelo autor é de que a sorte tem grande parcela de culpa nesse processo de conquista.

Nessa mesma linha, Hendel et al. (2015) analisou os dados de uma forma diferente. Utilizando diversas demonstrações matemáticas, os autores descobriram que o jogador atacante deve ter, em tropas, 86% a mais do que o oponente mais 2 tropas para ter 50% de chance de conquistar o território. Sabendo destas chances, é possível seguir para questões mais psicológicas de estratégias, como definir perfis de jogadores.

Por fim, jogos, de uma forma geral, são muito utilizados no processo de aprendizagem, independente de tema ou matéria. O jogo Risk tem um contexto histórico e geográfico interessante, podendo ter seu tabuleiro modificado para expressar informações reais de comércio ou alianças entre os territórios, alterando fronteiras, tamanho do continente e quantidade de territórios em cada continente. Alguns trabalhos utilizam os jogos Risk e War com o objetivo de ensinar sobre geopolítica (ARAÚJO et al., 2018), geografia (ROEHRS, 2017), política internacional (MARKS, 1998) e *design* (MEDEIROS; GONÇALVES, 2013). Da mesma forma, existem versões que representam os mais diversos períodos históricos, como o Risk Europe, com o mapa representando a era medieval na Europa, e o Risk Godstorm, com temática mitológica².

Com isso, é possível entender que Risk e War não são jogos largamente explorados nesses quesitos, nem possuem um método ótimo, preciso e fixo em nenhuma área apresentada nesta seção. Quando focamos nos métodos computacionais, a grande maioria das pesquisas foca em criar uma IA capaz de jogar Risk. Alguns métodos foram testados, porém, existem outras técnicas e combinações ainda a serem exploradas. Da mesma forma, diversas estatísticas podem ser analisadas, como a quantidade de movimentos possíveis ou a probabilidade de conquistar um território. Por mais que existam trabalhos que desenvolvem esta área, outras métricas podem ser valorizadas também, como balance-

²<https://www.ultraboardgames.com/risk/editions.php>

amento de mapas e tropas ganhas. Contudo, quando nos afastamos dessas ideias, mas ainda na mesma área, a quantidade de trabalhos envolvendo Risk e War se torna escassa.

3.2 Geração automatizada de jogos

A geração automática em jogos tem seu foco na criação de conteúdos específicos, como mapas de vídeo games (SNODGRASS; ONTAÑÓN, 2017) e cartas de jogos de cartas colecionáveis (CHEN; GUY, 2020). Estes trabalhos tem como objetivo a automatização do processo de criação de um ou mais elementos do jogo em específico, mas conservam todo o ambiente restante e, principalmente, as regras originais.

Quando falamos na criação automática de um jogo inteiro, incluindo as regras, vemos que os desafios mudam. Como uma das primeiras tentativas de evoluir as regras de um jogo, Togelius and Schmidhuber (2008) constrói uma base sólida de geração de um jogo inteiro, além também de conceitos de diversão e avaliação. São definidos parâmetros e valores que estes conceitos podem assumir dentro do estilo de jogo *Pac-man*. Grande parte dos jogos não são jogáveis, por questões de tempo e vencibilidade. Contudo, foram gerados jogos possíveis de jogar, porém, em sua maioria, não são muito divertidos ou extremamente desafiadores. O objetivo deste trabalho não é gerar jogos ótimos logo na primeira tentativa, mas sim mostrar que é possível e que existe uma nova área a ser explorada.

Também utilizando técnicas evolutivas, o trabalho de Hom and Marks (2007) utiliza ambos GGP e ZRF da *Zillions of Games*³ na busca para gerar jogos balanceados, ou seja, sem vantagem sobre quem joga primeiro. São variados o tamanho do tabuleiro, o tipo de peça e a condição de vitória, totalizando 144 possíveis jogos. Seguindo o objetivo de gerar jogos sem empates, o processo obteve considerável sucesso. Além disso, tem como trabalho futuro a avaliação de jogadores humanos, a fim de testar o quanto esses jogos são interessantes e desafiadores.

Tendo como objetivo criar uma biblioteca de mini jogos para jogos mais complexos, Ashlock (2010) trabalhou na geração evolutiva automática de dois tipos de *puzzles* de labirinto. O parâmetro modificado durante o processo é unicamente o tabuleiro, variando a posição de obstáculos, barreiras e objetivo. Um algoritmo de programação dinâmica mede o *fitness* como a quantidade mínima de movimentos necessários para completar o jogo. Assim, a avaliação maximiza essa quantidade mínima, gerando jogos simples, po-

³<https://www.zillions-of-games.com/>

rém com soluções complexas. Com essa avaliação, o processo gerou diversos tabuleiros nos mais diferentes níveis de dificuldade, porém sem avaliação de humanos ou baseada em métricas.

Utilizando uma Linguagem de Descrição de Video Games, (NIELSEN et al., 2015) buscam expandir a geração automática de jogos para video games. O processo de geração é feito por meio de um algoritmo evolutivo e o *fitness* de cada jogo é medido utilizando a diferença de desempenho de jogadores bons e jogadores ruins. O princípio é de que jogos bons fazem com que jogadores também bons joguem melhor do que os ruins. Por fim, os autores concluem que a geração de video games foi parcialmente bem-sucedida, pois, ao mesmo tempo que as ideias geradas são boas, a jogabilidade não foi interessante. Como nos trabalhos acima, a avaliação de humanos é o trabalho futuro de maior importância.

É importante ressaltar que nenhum trabalho mencionado nesta seção tem o objetivo de substituir *designers* humanos. Ao invés disso, visam encontrar novas mecânicas de jogo e novos gêneros, bem como servir de ferramenta de apoio para os *designers* humanos. Só um humano tem a capacidade de criar novas regras e novos elementos, o computador tem a capacidade de recombinar as regras já existentes, não agindo fora do seu domínio (BROWNE, 2011).

Todos os autores destacam a importância de aplicar os métodos apresentados por eles em outros jogos e com diferentes objetivos. Também, é demonstrado o interesse em gerar automaticamente novas regras, tabuleiros e peças, e não utilizar somente os já definidos. Por fim, embora nenhum trabalho utilize validação humana, esse é o teste final e também o mais importante, visto que jogadores humanos são os principais interessados em ter jogos novos e divertidos.

Por fim, o presente trabalho, no contexto de geração automática de jogos, se baseia principalmente no trabalho de Browne (2011), conforme apresentado na Seção 2.3. Em seu livro, Browne relata todo o processo de criação da Ludi e seus testes com jogadores humanos, o principal ponto faltante nos demais trabalhos apresentados nesta seção. Esses testes foram importantes para calibrar os valores das métricas e a sua importância dentro do *pipeline*. O principal resultado da Ludi foi o Yavalath, primeiro jogo criado por um computador a ser publicado.

Porém, o *pipeline* definido por Browne (2011) aborda uma quantidade limitada de jogos de tabuleiro. Por isso, baseado no trabalho de Browne, Bombardelli (2022) utiliza o processo de design evolutivo de jogos na criação de novas variantes do Risk, um jogo de

mesa que possui, além do tabuleiro, dados e cartas. Estes equipamentos trazem elementos a mais no *pipeline*, como a aleatoriedade. O principal resultado obtido foi a geração de novos mapas de Risk, mas que, contudo, possuem pouca influência na qualidade de uma partida.

No entanto, a versão utilizada por Bombardelli (2022) possui algumas limitações. A versão do Risk implementada contém simplificações, como a falta de cartas e poucos parâmetros de evolução. Além disso, o *pipeline* considera poucos critérios de avaliação, fazendo com que a nova versão criada não seja avaliada da forma mais completa possível.

Com isso, o trabalho de Bombardelli (2022) completa uma lacuna identificada na pesquisa de Browne (2011) sobre o *design* evolutivo de jogos de mesa. Porém, outras limitações foram criadas e devem ser solucionadas, como uma versão mais completa do Risk, adição de mais parâmetros de evolução e de métricas de avaliação.

Parte importante deste processo de geração automática é a plataforma de testes dos jogos. A Lux Delux⁴, da empresa Sillysoft, é amplamente utilizada principalmente pela sua grande variedade de agentes já implementados e pela facilidade em incluir novos. Diversos trabalhos (WIKLUND et al., 2015; FERRARI; ASSUNÇÃO, 2022; GIBSON; DESAI; ZHAO, 2010) relatam testes bem sucedidos utilizando a plataforma. Contudo, apesar de ser amplamente utilizada, a plataforma apresenta algumas limitações dentro do escopo do presente trabalho. A principal adversidade é a mudança das regras. Como não se trata de um *software* livre, não há acesso livre ao código do jogo, não possibilitando as mudanças necessárias para realizar os testes.

O *framework* TAG (Tabletop Games) (GAINA et al., 2020) foi desenvolvido com o objetivo de fornecer uma estrutura capaz de abranger diferentes tipos de jogos modernos em uma plataforma comum. Os jogos devem ser incorporados com base nos conceitos definidos pelos autores de estado e lógica de jogo. Contudo, ampliar o *framework* para fazer parte do processo de *design* evolutivo seria complexo, visto que a aquisição de métricas é executada de forma diferente, por exemplo. Além disso, a forma como a definição dos jogos é feita no TAG resultaria em partidas com muitos estados possíveis, aumentando consideravelmente o tempo de execução, sendo que o GA requer rapidez na etapa de testes.

⁴<https://sillysoft.net/lux/>

3.3 Conclusões

Ao longo deste capítulo foi possível entender como o Risk é explorado em diferentes áreas. Porém, sua principal aparição se dá no desenvolvimento de uma IA capaz de jogar o jogo. Grande parte dos esforços se concentram nesta área, mas também abrangem a análise de estatísticas e a utilização em ambiente educacional.

Além disso, este capítulo também menciona a geração automática de conteúdo. A grande maioria dos trabalhos procura gerar somente elementos específicos de um jogo, preservando o restante. As pesquisas em torno da geração automática de jogos inteiros são iniciais e em menor quantidade se comparadas com a geração de um conteúdo específico. O trabalho de Browne (2011), nesse sentido, cria uma base sólida, mas não abrange jogos de mesa em seu escopo.

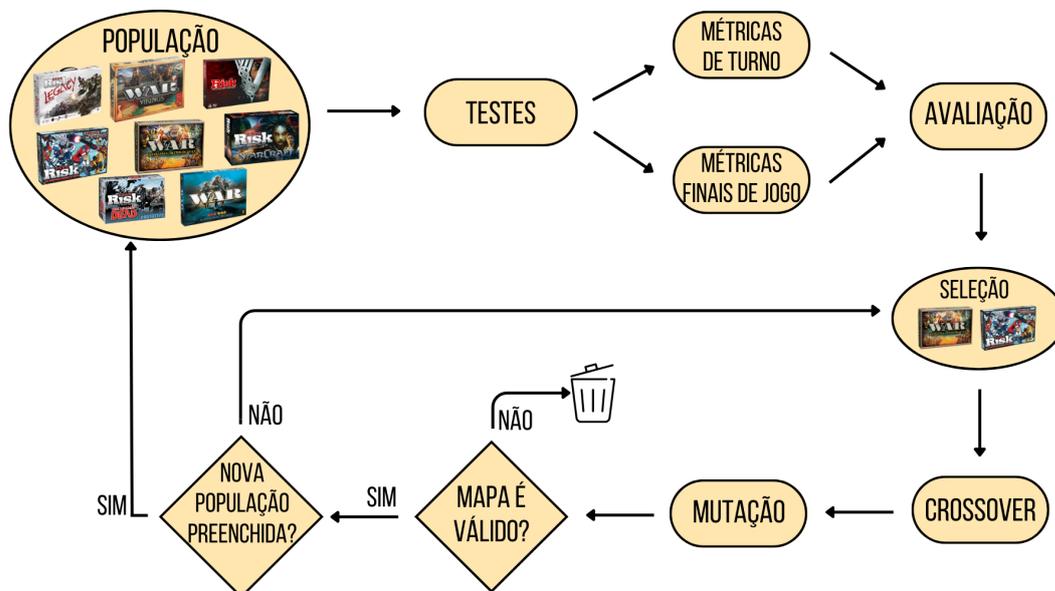
As plataformas de jogos de mesa apresentadas neste capítulo conseguem englobar diversos tipos de pesquisa, como testes com agentes. Porém, quando é necessário modificar as regras do jogo, ou seja, o programa em si, e executar as partidas e ter respostas dos agentes de forma ágil, elas não são suficientes.

Portanto, o presente trabalho se caracteriza como uma sequência da pesquisa realizada por Bombardelli (2022), realizando modificações e evoluções. Da mesma forma, explora de maneira pioneira a criação automatizada de versões dos jogos Risk e War.

4 DESIGN EVOLUTIVO DE VARIANTES DE RISK

O processo de geração automática de versões do Risk é capaz de criar, jogar e avaliar as versões geradas. Todo este processo é apresentado na Figura 4.1, onde a base é um algoritmo genético como um problema de minimização, sendo que o objetivo é encontrar o indivíduo com o menor *fitness* possível. São adicionadas etapas de teste e avaliação, devido à natureza do trabalho. A implementação foi feita na linguagem Python e se encontra disponível de forma aberta no GitHub¹.

Figura 4.1 – Pipeline implementado na geração de versões do jogo Risk. Cada nova população preenchida representa uma geração. A execução termina quando o número de gerações estabelecido for atingido.



Fonte: A Autora.

A versão do Risk utilizada neste processo é apresentada na Seção 4.1. A representação deste jogo como indivíduo do processo evolutivo é feita na Seção 4.2. Após isso, o modo como o GA foi implementado é apresentado na Seção 4.3, bem como definições importantes das operações genéticas. Por fim, a Seção 4.3.4 especifica como as versões são testadas e a Seção 4.3.5 apresenta a avaliação das mesmas.

¹<https://github.com/lanabr/Risk-Generation>

4.1 Risk simplificado

Neste *pipeline*, uma versão mais simples do Risk apresentado na Seção 2.2 foi utilizada. Esta variante é baseada na versão apresentada no trabalho de Bombardelli (2022). Porém, no presente trabalho, algumas modificações foram feitas para tornar esta versão mais próxima da versão original.

A maior parte das regras expostas na Seção 2.2 foram mantidas. A fase de alocação depende do parâmetro escolhido durante a evolução das regras do jogo. A troca de cartas de território também foi mantida. Quando o jogador possuir 5 cartas, ele é obrigado a realizar a troca. Contudo, as cartas de cada jogador não são mantidas escondidas. Isso foi feito com o objetivo de manter a versão simplificada.

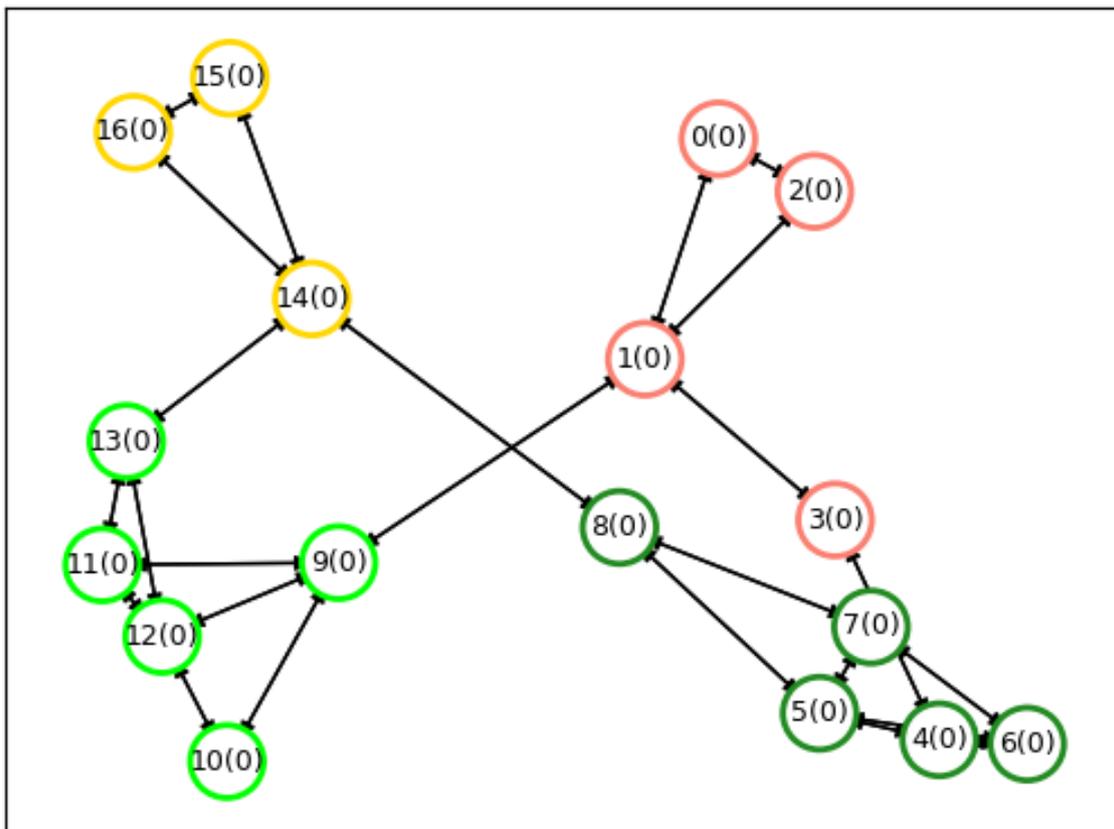
Durante a fase de batalhas, os ataques e defesas ocorrem como apresentado nas regras e também de acordo com os parâmetros escolhidos para a variação. Na fase de fortificação, os jogadores também tem plena liberdade de escolher as mudanças a serem realizadas.

A principal diferença para a versão original é a omissão das cartas de objetivo. Esse tipo de carta romperia com a característica de informação perfeita dos jogos combinatoriais, uma vez que o objetivo é um elemento privado e único de cada jogador. Por isso, em todas as variações possíveis, o objetivo de todos os jogadores é conquistar todos os territórios do mapa.

Devido ao ambiente digital, não foi necessário implementar os três modos de representação de tropas, armazenando simplesmente a contagem delas. Como no ambiente físico há uma limitação de espaço, seria impossível representar a quantidade de tropas ao final do jogo com equivalência absoluta nos territórios. Um território pequeno não comporta, por exemplo, 20 peças de Infantaria, mas permite duas peças de Artilharia naturalmente. Também, a implementação permite somente 2 jogadores. É possível adaptar a implementação para suportar até 6 jogadores, porém foge do escopo deste trabalho.

É possível visualizar os mapas criados e o andamento de uma partida através de uma *engine* simples. A Figura 4.2 mostra o mapa inicial, sem nenhuma alocação de tropas, onde os círculos representam os territórios e as linhas representam as fronteiras entre eles. Os números representam a identificação dos territórios, que vai de 0 ao número total de territórios menos um. Os números entre parenteses indicam a quantidade de tropas no território. As diferentes cores que contornam os círculos representam o agrupamento de territórios em um continente, ou seja, cada cor representa um continente.

Figura 4.2 – Visualização do mapa no início de uma partida.



Fonte: A Autora.

A Figura 4.3 mostra o mapa logo depois da fase de alocação de tropas. Cada cor que preenche os círculos identifica um jogador. É possível perceber também que o número entre parênteses mudou, indicando a adição de pelo menos uma tropa no território.

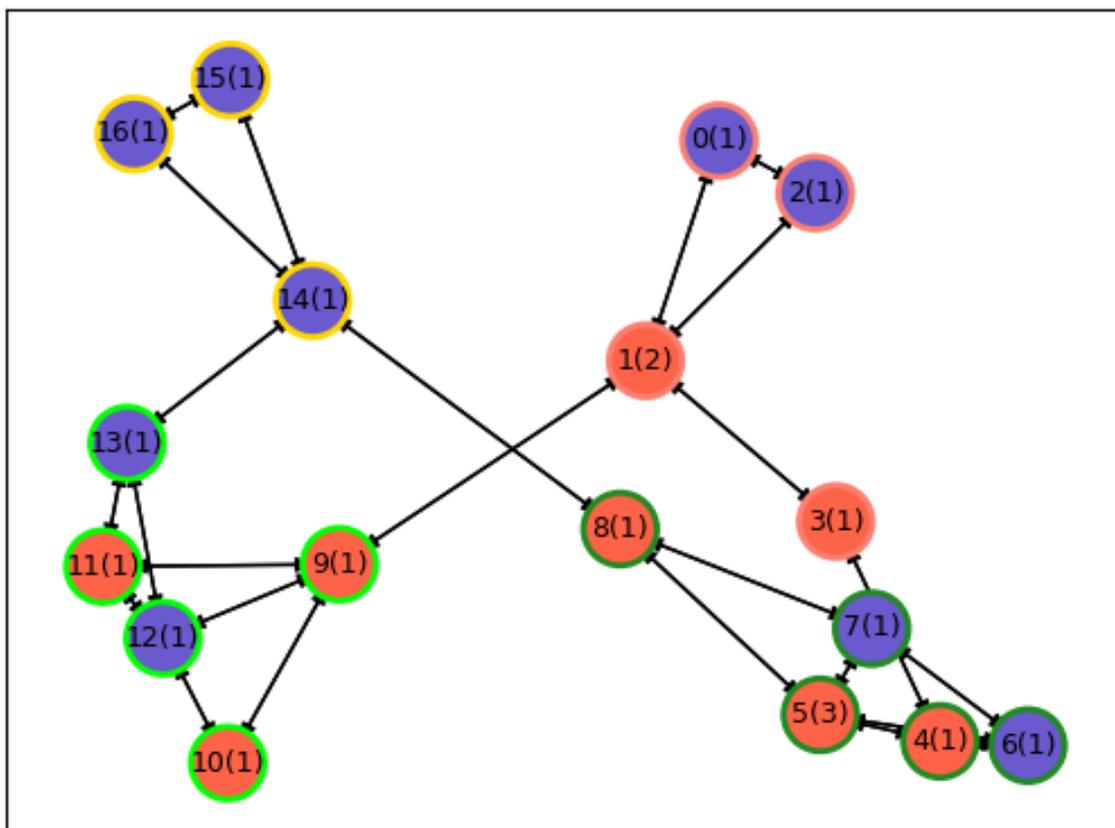
Por fim, também é possível jogar sendo um jogador humano. A escolha das ações, neste caso, é feita por linha de comando. As instruções e os comandos utilizados durante uma partida com um jogador humano podem ser vistos no Apêndice A.

4.2 Parâmetros

Conforme apresentado na Seção 2.2.2, as diferenças entre os jogos Risk e War foram as motivações para encontrar detalhes que podem ser mudados para tentar melhorar o jogo. Além das mudanças feitas no mapa, como remoção e adição de territórios, já discutidas por Bombardelli (2022), este trabalho busca customizar mais 4 aspectos:

- Modo de distribuição inicial de territórios;

Figura 4.3 – Visualização do mapa após a alocação de tropas.



Fonte: A Autora.

- Quantidade de dados defensivos dos jogadores;
- Quantidade de tropas movidas na conquista de um novo território;
- Fator de quantidade de tropas recebidas no início de cada turno.

O modo de distribuição inicial de território pode variar de duas formas: os territórios são designados de forma aleatória para os jogadores ou os jogadores escolhem alternadamente os territórios de acordo com a sua estratégia pessoal. Este parâmetro pode fazer com que os jogadores já tenham um continente completo em sua posse desde a primeira fase de ataque.

A quantidade de dados defensivos dos jogadores pode variar entre 2 ou 3 dados. De certa forma, defender com 2 dados dá vantagem para o jogador atacante, uma vez que ele tem uma chance a mais de tirar um número maior do que o defensor nos dados. Por consequência, defender com 3 dados dá uma certa vantagem para o jogador defensor, visto que agora a batalha foi equiparada e o defensor tem vantagem quando há empate.

A quantidade de tropas movidas na conquista de um novo território pode influenciar diretamente o andamento das partidas. Esse parâmetro pode variar entre mínimo

ou máximo, sendo que o mínimo é a quantidade de tropas referentes às batalhas ganhas no combate menos uma e o máximo é a quantidade de tropas pertencentes ao território atacante menos uma. Se mover a quantidade máxima, o jogador pode seguir atacando e alcançar um território mais distante, criando um corredor de conquistas no meio de territórios inimigos. Por outro lado, quanto mais longe for, mais fracos ficarão os territórios recém-conquistados. Movendo a quantidade mínima, o atacante não consegue ir muito longe, só consegue atacar territórios próximos. Porém, os poucos territórios conquistados ficarão protegidos.

A quantidade de tropas recebidas no início de cada turno é o número de territórios dividido pelo fator, sendo que o fator pode variar entre 1, 2, 3 e 4. É importante lembrar que o número de tropas recebidas no início de cada turno é o valor mínimo entre 3 e o valor calculado pela fórmula.

Por fim, o mapa pode variar de diversas formas. Quando o *crossover* é feito, partes dos mapas dos pais são fundidas e formam um mapa com características de ambos. É nesse processo que a quantidade de territórios pode mudar, quando um dos pais possui um número muito diferente de territórios que o outro. Após isso, na mutação, existe a possibilidade de retirar, adicionar ou mover um território, além de trocar o valor das tropas entre dois continentes ou adicionar 1 tropa a esse valor.

O mapa resultante desse processo precisa seguir algumas regras, devendo ser planar, para representar um mapa, e conexo, para que todos os territórios sejam alcançados. Essa verificação foi feita utilizando a biblioteca Networkx e os métodos *is_planar*² e *is_strongly_connected*³. A planaridade é verificada pelo Teste de Planaridade Esquerda-Direita (BOYER et al., 2003) e a conectividade é verificada com um algoritmo de busca em profundidade.

A maior influência de um mapa pode ser vista na duração de jogo. Quanto maior o mapa, mais demorado será para jogá-lo. É importante ressaltar que não existe limite para o crescimento do mapa. O grafo pode expandir sem quaisquer restrições de tamanho. Portanto, o espaço de busca para a otimização é infinito.

Um resumo dos valores que os parâmetros podem assumir durante o *pipeline* é apresentado na Tabela 4.1. As possibilidades dos mapas não estão expressas, pois podem assumir infinitos padrões.

²https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.planarity.is_planar.html

³https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.components.is_strongly_connected.html

Tabela 4.1 – Valores possíveis de cada parâmetro.

Parâmetro	Valores Possíveis
Modo de distribuição inicial de territórios	Escolha ou distribuição aleatória
Quantidade de dados defensivos dos jogadores	2 ou 3
Quantidade de tropas movidas na conquista de um novo território	Mínimo ou Máximo
Fator de quantidade de tropas recebidas no início de cada turno	1, 2, 3 ou 4

Fonte: A Autora

4.3 Algoritmo Genético

O algoritmo genético utilizado neste trabalho segue o fluxograma apresentado na Figura 4.1. A população inicial é criada e passa pelos processos de seleção, reprodução e substituição. Durante a reprodução, os grafos dos mapas recém criados podem apresentar erros difíceis de encontrar ou resolver. No entanto, alguns são simples e são resolvidos antes de passar para a etapa seguinte. O critério de parada utilizado é o número de gerações executadas.

O cromossomo que representa um jogo é uma lista de parâmetros, exceto o mapa que não é representado explicitamente. Ao invés disso, há um caminho do arquivo JSON que possui a sua descrição, e esse arquivo é interpretado para os operadores genéticos operarem sobre os mapas.

4.3.1 População Inicial

A criação da população inicial se dá de forma aleatória. Os valores dos parâmetros de cada indivíduos são escolhidos entre as possibilidades pré-estabelecidas apresentadas na Tabela 4.1. Dentre os mapas disponíveis para a criação inicial estão 5 mapas oficiais e 5 mapas artificiais. Os mapas oficiais, apesar de serem publicados e balanceados, são muito grandes. Todos contém 42 territórios divididos em 6 continentes, normalmente deixando

as partidas extremamente longas⁴. Os 5 mapas foram retirados de versões diferentes do jogo Risk, procurando criar uma variedade de opções, sendo eles:

- Mapa 1: Risk
- Mapa 2: Risk: Starcraft
- Mapa 3: Risk: The Transformer Editions
- Mapa 4: Star Wars Risk: The Clone Wars Edition
- Mapa 5: Risk: Halo Wars

Os 5 mapas criados artificialmente são menores, tendo entre 9 e 17 territórios. Eles têm o objetivo de criar indivíduos mais diversos, forçando a criação de versões de mapas mais compactos. Assim, outras situações serão exploradas, trazendo mais variedades nos jogos gerados. Os mapas do War não foram utilizados para compor a base inicial, pois possuem menor variação nas demais versões e o mapa básico do War é muito parecido com o Mapa 1, o mapa básico do Risk. Essas semelhanças serão discutidas na Seção 5.5.

4.3.2 Seleção

O modo de seleção testado foi o torneio, porém com algumas modificações da versão especificada na Seção 2.4.1. Dado um tamanho para o grupo do torneio, indivíduos da população são escolhidos aleatoriamente para participar. Depois disso, os dois indivíduos com os menores *fitness* são escolhidos para serem os pais de dois novos indivíduos.

Além disso, no início de cada iteração ocorre o elitismo, onde o indivíduo com o menor *fitness* da antiga população é mantido na nova. Isso garante que todo o trabalho feito até o momento não seja perdido ou esquecido.

4.3.3 Crossover e Mutação

As operações de *crossover* e mutação são divididas em duas partes: os atributos e o mapa. Os atributos são recombinados com uma máscara aleatória, mas que sempre separa metade do indivíduo. Metade de um indivíduo pai é combinada com a metade

⁴Existem fóruns na internet que discutem os problemas do Risk, sendo que um dos mais comentados é a duração de uma partida, como em: https://www.reddit.com/r/boardgames/comments/2ap923/what_is_the_problem_with_risk/. Além disso, em testes realizados posteriormente, podemos ver que a métrica de duração do Risk tem valor 0.99, sendo 0 o mínimo e 1 o máximo.

complementar do outro indivíduo pai gerando um filho, e o inverso gera o segundo filho. Durante a mutação dos atributos, se a taxa de mutação for alcançada, o novo valor do parâmetro é alternado. Cada mutação ocorre independentemente, mas com a mesma probabilidade. Uma mutação ocorrer não significa que todos os outros parâmetros sofrerão mutação também, mas pode ocorrer mais de uma mudança no mesmo indivíduo. Isso foi feito porque as operações de mutação não são invasivas, não retiram e nem adicionam territórios. A operação de *crossover* possui total responsabilidade no tamanho dos mapas.

Já com o mapa, as operações se tornam mais complexas. No *crossover*, são escolhidos alguns continentes de ambos os mapas para serem combinados conforme eles se encontram originalmente, apenas criando conexões para manter a coerência. Todos os territórios e conexões originais são mantidos durante a ligação. Durante a mutação, algumas operações podem ser realizadas, sendo elas:

- Criação de nova conexão;
- Remoção de uma conexão;
- Movimentação de um território de um continente para outro;
- Troca do valor de bônus entre dois continentes;
- Modificação do valor de bônus de continente em 1 unidade.

Cada mutação tem a sua probabilidade de acontecer, isto é, uma mutação ocorre sem a necessidade de acontecer as outras. Além disso, a taxa de mutação é um hiperparâmetro.

4.3.4 Testes do Jogo

Após a evolução dos parâmetros, é preciso testar a jogabilidade da nova versão criada. Se a versão chegou até esta etapa, não há nenhum problema com o mapa gerado, visto que ele atende aos requisitos impostos. Porém, a combinação de todos os parâmetros pode resultar em um jogo muito lento, onde ambos os jogadores têm acesso a uma quantidade demasiada de tropas, tornando-se impossível a conquista de um território.

Diversas partidas são jogadas a fim de testar a maior quantidade de casos possíveis de escolha e rumo que uma partida pode seguir. Um dos principais elementos da fase de testes é o jogador, uma vez que ele tem grande responsabilidade na qualidade dos jogos, visto que as métricas calculadas posteriormente são baseadas nas ações tomadas por ele.

4.3.4.1 Agente baseado em regras

Como relatado por Bombardelli (2022), o desenvolvimento de um agente utilizando um MCTS básico não foi bem sucedida. O número de estados a serem processados é muito grande e por isso o MCTS não toma boas decisões com tempo limitado. Considerando este contexto, um simples agente baseado em regras foi desenvolvido em Bombardelli (2022) e será utilizado também nesta dissertação. Este agente também foi atualizado conforme as revisões feitas na versão utilizada do jogo Risk.

Durante a fase de alocação, o agente procura colocar suas tropas nos territórios do continente que retorna o maior valor de tropas. Assim, se conquistar esse continente logo nesta primeira fase, já terá o maior bônus de tropas logo na primeira fase de ataque. Se não há nenhum continente vazio, o agente escolhe de forma aleatória. Quando há bônus de continente, as tropas são colocadas onde há menos unidades, mas sempre nas fronteiras do próprio continente que forneceu o bônus.

Já na troca de cartas, o agente sempre procura uma carta de um território que ele possua. Assim, consegue tropas a mais para ele. Se possui 3 ou 4 cartas e não possui carta com território o qual seja o dono, a troca não é feita. Porém, se possui 5 cartas, a troca é feita obrigatoriamente, pelas regras do jogo.

Quando ataca, o agente busca por batalhas que tenha vantagem numérica, ou seja, busca atacar territórios com menos tropas do que o seu território, tentando conquistar o máximo possível de territórios enquanto tem tropas para isso. Se não há situações em que isso acontece ou todas as opções já esgotaram, o agente opta por passar o turno ao invés de continuar tentando. Esse comportamento é cuidadoso, uma vez que escolhe batalhas em que possui maior oportunidade de ganhar.

Como última movimentação do turno, quando tem a oportunidade de mover suas tropas e fortificar outros territórios, o agente opta pela mesma estratégia da fase de alocação. Com isso, ele move tropas para territórios menos populosos onde há fronteira com o oponente. Se não existe fronteira inimiga, o território pode permanecer com apenas uma tropa.

Com essas características, é possível jogar o Risk de forma sensata, sem nenhuma estratégia mais sofisticada. Os dois jogadores da fase de testes utilizam o mesmo agente. O agente descrito é rápido, facilitando a avaliação de muitos jogos, o que é uma necessidade do GA.

4.3.5 Avaliação

Medir o quanto um jogo é divertido, se é interessante ou se faz o jogador se entreter podem ser tarefas difíceis, pois cada jogador tem a sua noção de diversão e entretenimento. Porém, é possível avaliar algumas características relacionadas à qualidade de um jogo na percepção dos jogadores.

Conforme visto anteriormente, na Seção 2.3, Browne (2011) apresentou 57 critérios de estética para medir a qualidade de um jogo. Divididos em critérios intrínsecos e extrínsecos, eles são utilizados com o objetivo de classificar os jogos gerados que, em sua maioria, são jogos completamente desconhecidos.

O jogador utilizado na fase de testes tem grande influência na fase de avaliação, uma vez que ele é quem joga pela primeira vez os jogos gerados. Se o jogador for mais agressivo, por exemplo, o *pipeline* criará uma tendência para jogos que permitam ao jogador ser mais agressivo.

Tendo como base as métricas definidas por Browne, para este trabalho, os critérios intrínsecos, medidos com base nas regras e equipamentos, não se fazem necessários, visto que são geradas somente versões de um mesmo jogo. Por isso, essas métricas teriam o mesmo valor para todos os jogos.

Já os critérios de viabilidade, medidos pelas consequências das jogadas de cada jogador, são extremamente úteis para definir se um jogo pode ser jogado sem falhas graves e de forma balanceada. São mais fáceis de medir que os critérios de qualidade e podem carregar tanta informação quanto eles.

Por fim, os critérios de qualidade, que medem a participação dos jogadores, tendem a ser mais complexos, porém carregam mais informação sobre o desenrolar da partida e sobre como os jogadores interagem entre si. São menos precisos que os critérios de viabilidade, porém tem maior relação com o engajamento dos jogadores.

Considerando o jogo Risk e suas características, foram escolhidas 7 métricas que favorecem o desaparecimento de jogos ruins, sendo 3 de viabilidade e 4 de qualidade. Dessa forma, é possível controlar a duração e o equilíbrio do jogo, além de garantir que haja algum tipo de emoção, mas sem torná-lo complexo demais para o jogador. Algumas dessas métricas precisam da avaliação do jogador a cada turno no tabuleiro.

A avaliação do tabuleiro para cada jogador p é uma heurística H_p que consiste em utilizar o número de territórios e tropas em conjunto (BOMBARDELLI, 2022).

$$H_p = \frac{T_p}{T_{total}} * w_{terr} + \frac{U_p}{U_{total}} * w_{unit} \quad (4.1)$$

Para isso, são definidos pesos w_{terr} e w_{unit} que representam a importância dos territórios e das unidades, respectivamente, no mapa de uma partida. Também, são calculadas as proporções utilizando o total de territórios T_p do jogador e o total de territórios T_{total} do mapa, além da quantidade de tropas U_p do jogador e o total de tropas U_{total} de todo o mapa.

As métricas são descritas nas Seções 4.3.5.1 até 4.3.5.7 a seguir.

4.3.5.1 Completude

É esperado que um jogo resulte mais vezes em vitórias, independente de qual jogador, do que em empates. Não é interessante entrar em uma partida sabendo que existem grandes chances de ela terminar em empate, afinal um dos maiores incentivos é terminar com a vitória sobre o oponente.

Assim, a completude (C_{comp}) é calculada em razão das partidas resultadas em vitória para os jogadores 1 ou 2 pela quantidade total de jogos J jogados na fase de testes (BROWNE, 2011):

$$C_{comp} = \frac{vitorias_1 + vitorias_2}{J} \quad (4.2)$$

O melhor resultado de C_{comp} é igual a 1, refletindo em um jogo onde sempre há um vencedor. Assim como em todas as métricas posteriores, a completude total é uma média de todos os jogos J jogados na fase de testes.

4.3.5.2 Duração

Tanto o War quanto o Risk podem parecer intermináveis, mas uma partida sempre vai terminar. Pode ser em 1 hora, 3 ou até 4 horas, mas sempre há a possibilidade de haver um vencedor. Porém, não é desejável que uma partida dure tanto tempo, porque assim pode se tornar cansativa e monótona, ao contrário das qualidades esperadas das versões a serem geradas.

A duração (C_{dur}) é calculada levando em consideração a duração preferida (D_{pref}) e a duração de cada uma das partidas jogadas (D_j) na fase de testes, que contém J partidas (BROWNE, 2011).

$$C_{dur} = \frac{1}{J} \sum_{j=1}^J \frac{|D_{pref} - D_j|}{D_{pref}} \quad (4.3)$$

A duração preferida foi definida como 24, sendo 12 turnos por jogador. Supondo que um jogador humano completa todas as fases do seu turno em 3 minutos, isso resulta em uma partida com 72 minutos. O resultado de C_{dur} pode variar de 0 a 1, sendo que o melhor valor é 0, significando que o conjunto de partidas analisadas não variou em relação ao valor preferido.

4.3.5.3 Vantagem

Um jogo não deve favorecer o primeiro ou o segundo a jogar. Se o jogador começa uma partida sabendo que está em desvantagem só porque é o segundo a jogar, aquela partida já se tornou desinteressante. É preciso manter um balanceamento, a fim de manter o engajamento de ambos os jogadores.

A vantagem (C_{vant}) é dada pela taxa de vitórias do jogador 1, ou seja, do primeiro a jogar (BROWNE, 2011):

$$C_{vant} = \frac{|vitorias_1 - (vitorias_1 + vitorias_2)/2|}{(vitorias_1 + vitorias_2)/2} \quad (4.4)$$

Se o resultado for acima de 0,5, significa que o primeiro a jogar tem vantagem sobre o segundo. Já se o resultado for abaixo de 0,5, significa que a vantagem é do segundo a jogar.

4.3.5.4 Fator de ramificação

O número de movimentos possíveis por turno geralmente nos indica a complexidade de um jogo, ou seja, quantas possibilidades o jogador precisa processar para tomar a melhor decisão a cada turno. Se um jogo é muito complexo, pode se tornar muito difícil de aprender ou muito improvável de encontrar a melhor decisão. Por outro lado, jogos muito simples podem se tornar desinteressantes com facilidade.

Enquanto uma partida está em andamento, algumas métricas são medidas por turno. A quantidade de movimentos possíveis para o jogador é medida somente no início de cada fase. Essa medição não é feita a cada movimento propriamente dito do jogador, pois haveria uma sobreposição de possibilidades. Já a heurística do tabuleiro é medida somente no final do turno, representando a avaliação final do mapa após todas as decisões

do jogador.

A quantidade de movimentos possíveis é medida nas três fases da partida, quando o jogador deve adicionar novas unidades, atacar e mover suas tropas. A contagem de movimentos possíveis de adicionar as novas unidades M_{adc} é feita multiplicando as unidades disponíveis u_p pelo total de territórios T_p do jogador p .

$$M_{adc} = u_p * T_p \quad (4.5)$$

Durante a fase de ataque, a medição foi simplificada. Normalmente, seguindo as regras do Risk, é possível atacar com menos tropas do que o máximo possível. Por exemplo, se o jogador tem 4 tropas e prefere atacar com 2 ao invés de 3, que seria o máximo e mais aconselhável, ele pode. Porém, raramente essa decisão é feita, visto que diminui a chance de vitória. Por isso, serão considerados somente ataques com a quantidade máxima de tropas. Assim, o cálculo da quantidade de movimentos possíveis durante a fase de ataque M_{atq} é o seguinte:

$$M_{atq} = \sum_{terr=1}^{T_p} T_{fi}(terr) * \min(3, u_{terr} - 1) \quad (4.6)$$

Para todos os territórios T_p em posse do jogador atual p , é preciso multiplicar o número de territórios inimigos na fronteira $T_{fi}(terr)$ por $u_{terr}-1$ tropas, sendo 3 o limite, ou seja, se o jogador tem mais do que 4 tropas, o máximo que pode utilizar para atacar é 3.

Para medir os movimentos possíveis durante a fase de fortificação M_{fort} , é preciso multiplicar as unidades u_{terr} existentes no território $terr$ pelo número de territórios $T_{fa}(terr)$ que fazem fronteira e estão sob a posse do jogador analisado. Isso é feito para todos os territórios que o jogador possui. É preciso desconsiderar uma tropa nesta avaliação, uma vez que uma unidade deve permanecer no território para demonstrar a posse.

$$M_{fort} = \sum_{terr=1}^{T_p} T_{fa}(terr) * (u_{terr} - 1) \quad (4.7)$$

As Equações 4.5, 4.6 e 4.7 são de elaboração da autora.

Com isso, o fator de ramificação (C_{fr}) (BROWNE, 2011) é calculado conforme abaixo, onde $M(t_n)$ é o total de movimentos possíveis disponíveis para o atual jogador no turno t_n , sendo a soma $M_{adc} + M_{atq} + M_{fort}$, e T_j é o total de turnos jogados durante o jogo j . Também, o componente \log_{10} é dividido por 2 para definir um limite de aproximadamente 100 movimentos.

$$C_{fr} = \frac{1}{J} \sum_{j=1}^J \min \left(1, \log_{10} \left(\frac{\sum_{n=0}^{T_j-1} M(t_n)}{T_j} + 1 \right) \right) / 2 \quad (4.8)$$

O melhor valor de C_{fr} é igual a 0,5, ou seja, 50 possibilidades a cada movimento, representando um fator de ramificação médio. Porém, como visto anteriormente na Seção 1, o fator de ramificação do Risk é muito alto, praticamente impossível de alcançar este valor. Portanto, o objetivo de medir o fator de ramificação é tentar pelo menos minimizar o valor.

4.3.5.5 Drama

Quando um jogador está perdendo, ele deve ter pelo menos a esperança de que vai conseguir ganhar. Se, uma vez que ele está perdendo, não tiver a chance de ganhar, essa partida não vai mais ser interessante para ele. Uma das características de um bom jogo é conseguir manter os jogadores interessados e envolvidos com a partida, não deixar que desanimem. Como o principal objetivo de um jogo é ganhar uma partida, uma das formas de prender a atenção do jogador é alimentar nele a esperança de ganhar.

O drama (C_{dra}) é calculado levando em consideração a quantidade de vezes em que o jogador que está ganhando sofreu alguma queda (BROWNE, 2011).

$$C_{dra} = \frac{1}{J} \sum_{j=1}^J \left(\frac{se\ H_g(t_n) < H_p(t_n) \{ \sqrt{H_p(t_n) - H_g(t_n)} \}}{contar(H_g(t_n) < H_p(t_n))} \right) \quad (4.9)$$

Para isso, são utilizadas a avaliação heurística do ganhador $H_g(t_n)$ e a do perdedor $H_p(t_n)$ durante o turno t_n em questão. Assim, podemos medir a severidade de cada desvantagem sofrida pelo atual ganhador. O resultado de C_{dra} pode variar de 0 a 1, sendo que o melhor valor é 0,5. Uma partida com muito drama pode ser muito conturbada, enquanto uma partida sem drama pode parecer monótona e previsível.

4.3.5.6 Movimentos extremos (killer moves)

Alguns movimentos têm a característica de alavancar a posição de um jogador, fazendo-o ir de perdedor para ganhador do turno. É interessante ter a possibilidade de fazer movimentos extremos, visto que pode mudar drasticamente a avaliação dos jogadores naquele momento. Porém, quando a liderança muda em todo turno, a partida pode se tornar desgastante. No Risk, um movimento extremo engloba todas as ações de um turno, visto que uma só movimentação não tem poder de mudar drasticamente a liderança.

A taxa de movimentos extremos (C_{me}) é calculada conforme a equação abaixo (BROWNE, 2011).

$$C_{me} = \frac{1}{J} \sum_{j=1}^J [H_a(t_n) - H_o(t_n)] - [H_a(t_{n-1}) - H_o(t_{n-1})] \quad (4.10)$$

Então, para cada partida jogada na fase de testes, apenas a maior diferença entre as avaliações dos jogadores em turnos subsequentes é utilizada no somatório. Assim, o somente o movimento mais extremo é utilizado no cálculo. Para isso, é utilizada a heurística de avaliação do jogador atual $H_a(t_n)$ e do oponente $H_o(t_n)$ no turno t_n e também no turno anterior, respectivamente $H_a(t_{n-1})$ e $H_o(t_{n-1})$. O resultado pode varia entre 0 e 1, e o melhor valor é 0,5, onde a quantidade de movimentos extremos se manteve mediana.

4.3.5.7 Mudança de Liderança

De forma similar ao Drama, se uma partida não tiver mudança de liderança constante, o jogo acaba se tornando tedioso. Sem a variação na liderança, o jogador que começa ganhando vai terminar ganhando também, tornando a partida desinteressante para o que está perdendo, pois sabe que nunca terá a chance de virar o jogo.

A mudança de liderança (C_{ml}) é calculada quando há diferença entre os líderes de turnos subsequentes (BROWNE, 2011).

$$C_{ml} = \frac{1}{J} \sum_{j=1}^J \left(\frac{\text{contar}(\text{líder}(t_n) \neq \text{líder}(t_{n-1}))}{T_j - 1} \right) \quad (4.11)$$

É preciso contar a quantidade de vezes em que o líder do turno atual $\text{líder}(t_n)$ não é o mesmo que o líder do turno anterior $\text{líder}(t_{n-1})$. Também, consideramos somente os turnos não finais T_j-1 de todos os jogos J jogados na fase de testes. O resultado de C_{ml} pode variar de 0 a 1, sendo que o melhor valor é 0,5, onde a variação na liderança não é extrema.

4.3.5.8 Fitness

Para o *fitness*, a forma de cálculo testada foi baseada em distância. Neste método, são escolhidos valores considerados ótimos para os critérios de avaliação. Assim, é calculada a distância do indivíduo analisado em relação a um indivíduo ótimo. Os valores

ótimos podem ser vistos na Tabela 4.2.

Tabela 4.2 – Valores ótimos para os critérios de avaliação.

Critério	Valor Ótimo
Compleitude	1
Duração	0
Vantagem	0
Fator de Ramificação	0.5
Drama	0.5
Movimentos Extremos	0.5
Mudança de Liderança	0.5

Fonte: A Autora

Após calcular as métricas da versão gerada, o cálculo da distância é feito como mostrado abaixo. O *fitness* de uma determinada versão v é calculado pelo somatório do valor absoluto da subtração do critério i da versão c_{vi} pelo valor ótimo do mesmo critério ot_i , onde I é uma lista com todos os critérios.

$$fitness_v = \sum_{i=1}^I |c_{vi} - ot_i| \quad (4.12)$$

Este trabalho usa o termo *fitness* com uma semântica diferente do habitual, pois normalmente é desejada sua maximização, satisfazendo a metáfora evolutiva da sobrevivência do indivíduo mais apto. Aqui, porém, o fitness descrito induz um problema de minimização. É possível transformar o fitness para obter um problema de maximização, mas o trabalho segue com a ideia de minimização, mantendo a clareza da definição de fitness de acordo com a Equação 4.12.

4.3.6 Substituição

Como o objetivo deste trabalho é gerar versões melhores do jogo Risk e não somente uma versão ótima, o critério de parada se baseia no número de gerações. Por mais que, pelos critérios de avaliação, tenhamos uma versão do *fitness* ótima de referência, ela não será usada para determinar a parada do algoritmo. Da mesma forma, não é conhecida uma combinação dos parâmetros que alcance a versão ótima das métricas de avaliação.

Ao final de cada iteração, toda a população utilizada para gerar a nova é substituída. Assim, a nova população é constituída pelo melhor indivíduo da população anterior

mais a prole produzida por essa população.

4.4 Resumo

O *pipeline* apresentado neste capítulo começa com a descrição da versão do Risk utilizada. Algumas simplificações foram feitas, como a retirada das cartas de objetivo e a permissão de somente dois jogadores. O restante das regras, como a forma de atacar e posicionar tropas se mantém o mesmo.

Os parâmetros utilizados para alterar a versão base do jogo são 5 no total: o modo de distribuição inicial de territórios, a quantidade de dados defensivos, a quantidade de tropas movidas na conquista de um novo território, o fator de quantidade de tropas recebidas no início de cada turno e o mapa do jogo. Estes parâmetros foram escolhidos tendo como base as diferenças já existentes entre os jogos Risk e War e os valores que eles podem assumir são expostos na Tabela 4.1.

O algoritmo genético definido ao longo deste capítulo tem como objetivo gerar versões do Risk apresentada modificando somente os parâmetros escolhidos. Para isso, é preciso otimizar a função de *fitness* definida na Equação 4.12, calculada com base nas métricas. Ao todo, 7 métricas foram incorporadas este cálculo: completude, duração, vantagem, fator de ramificação, drama, movimentos extremos e mudança de liderança.

A seleção dos melhores indivíduos é feita considerando esta função de *fitness* e tem como objetivo otimizá-la. O *crossover* e a mutação são realizados em sequência, buscando a combinação dos parâmetros que resulta em um *fitness* melhor. O principal responsável pelos testes é o agente, também apresentado neste capítulo. Ele é baseado nas regras do jogo e toma decisões simples, porém objetivas. Ao final do processo descrito, a população resultante é composta por novas versões do jogo Risk.

5 EXPERIMENTOS

O primeiro passo dos experimentos foi encontrar os melhores valores para os hiperparâmetros do algoritmo genético, descrito na Seção 5.1. Após isso, na Seção 5.2, veremos como o mapa evoluiu e quais foram as principais características nas gerações finais. Também, na Seção 5.3, são apresentados os resultados das evoluções dos outros parâmetros de evolução.

Alguns testes de jogabilidade e com mapas dos jogos originais também foram realizados e são descritos na Seção 5.4. Nesta etapa, o objetivo é entender como o *pipeline* se comporta com conjuntos de parâmetros referentes aos jogos reais e até onde ele consegue chegar variando esses jogos. Além disso, um teste final foi feito com uma tentativa de gerar o War a partir das regras do Risk, conforme relatado em 5.5.

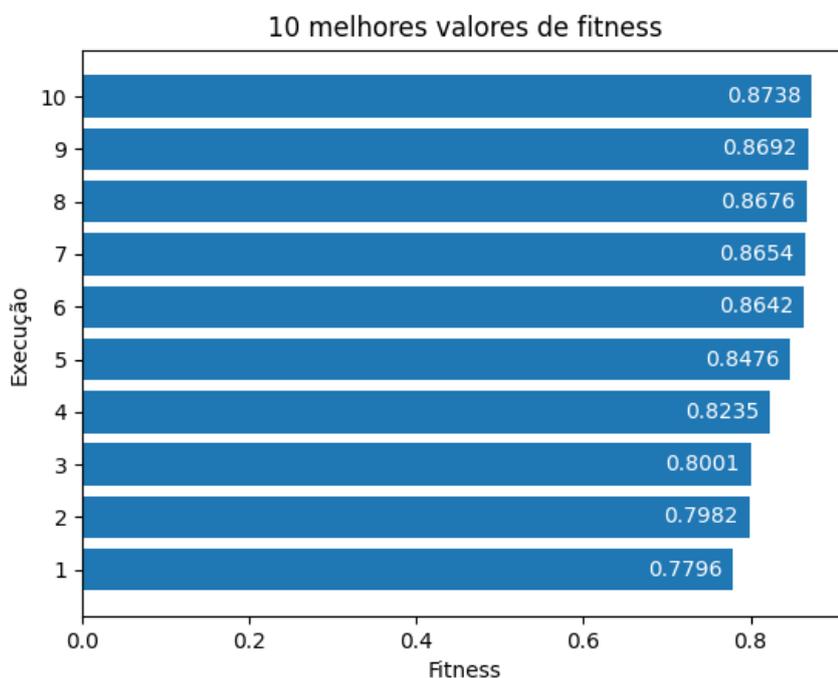
5.1 Hiperparâmetros

Durante esta primeira fase de testes, foram analisados diferentes valores de 4 hiperparâmetros. São realizadas 100 partidas para cada jogo gerado, sendo que o limite máximo de turnos é de 48 e o tempo máximo para cada partida é de 60 segundos. Estes hiperparâmetros foram pensados levando em consideração o tempo para a testagem dos jogos. Isso foi feito com o objetivo de encontrar a melhor combinação para futuros testes. Os parâmetros e os valores testados são os seguintes:

- Número de gerações: 10, 30, 50, 70, 90, 110, 130, 150, 170, 190. Partindo de 10 gerações, valor mínimo escolhido para gerar um resultado confiável, os valores foram crescendo ao passo de 20 gerações.
- Tamanho da população: 5, 10, 15, 20, 25, 30, 35, 40, 45, 50. Começando com 5, os valores foram aumentando ao passo de 5 até chegar em 50 indivíduos por geração.
- Tamanho do torneio: o número de participantes no torneio de seleção pode variar de 2 até a metade do número de indivíduos da população arredondado para baixo, ao passo de 2. Por exemplo, se a população for de 30, os números de participantes do torneio varia de 2 até 14.
- Taxa de mutação: 0,1, 0,2, 0,4, 0,6, 0,8. Começando por 0,2, a probabilidade de uma mutação ocorrer pode variar até 0,8, ao passo de 0,2. A taxa de 0,1 foi adicionada levando em consideração o parâmetro utilizado por Browne (2008).

Todas as combinações possíveis entre estes 4 hiperparâmetros foram testadas, a fim de avaliar e encontrar a melhor opção, ou seja, a que resultasse em um *fitness* mais próximo ao ótimo, de acordo com o cálculo baseado em distância. No total, foram 3.250 execuções distintas. A Figura 5.1 mostra o *fitness* das 10 melhores execuções, identificadas de 1 a 10, cujos hiperparâmetros se encontram na Tabela 5.1. É importante lembrar que estes valores foram retirados ao final da última geração de cada execução.

Figura 5.1 – *Fitness* resultante das melhores execuções.



Fonte: A Autora

Analisando os parâmetros que levaram a um *fitness* mais baixo, o número de gerações sendo o mais baixo foi curioso. Neste contexto, espera-se que um número baixo de gerações não seja suficiente para evoluir os indivíduos ao seu máximo. Para entender a situação, a Figura 5.2 mostra o *fitness* ao longo das gerações para a melhor (a) e segunda melhor (b) configurações de hiperparâmetros (configurações 1 e 2 da Tabela 5.1). É possível perceber que, na segunda melhor execução, houve bastante variação do menor valor de *fitness* de cada geração. Perto da geração 100, por exemplo, uma geração chegou muito perto do valor da última geração. Já na primeira melhor execução, como o número de gerações é bem reduzido, essa oscilação foi mais suave.

Quando o elitismo é aplicado na substituição de uma população, o melhor indivíduo da população anterior se mantém na composição da nova população. Quando isso acontece, o indivíduo é avaliado novamente. A avaliação é estocástica devido à aleatori-

Tabela 5.1 – Parâmetros de cada execução geradora dos 10 melhores *fitness*

Execução	<i>Fitness</i>	Número de gerações	Tamanho da população	Tamanho do torneio	Taxa de mutação
1	0,77956	10	50	22	0,6
2	0,79819	150	30	12	0,1
3	0,80013	50	50	16	0,6
4	0,82350	150	30	8	0,6
5	0,84763	150	20	6	0,8
6	0,86418	150	50	6	0,4
7	0,86544	30	45	2	0,6
8	0,86763	90	45	20	0,6
9	0,86920	30	35	6	0,8
10	0,87376	10	25	8	0,8

Fonte: A Autora

idade nos lances de dados. Isso pode fazer com que o indivíduo que antes era o melhor, acabe não tendo uma avaliação tão boa quanto antes. Analisando a Figura 5.2a, o *fitness* mínimo foi rapidamente encontrado e, como não houve tempo para sofrer as diversas alterações vistas na Figura 5.2b, ele se manteve.

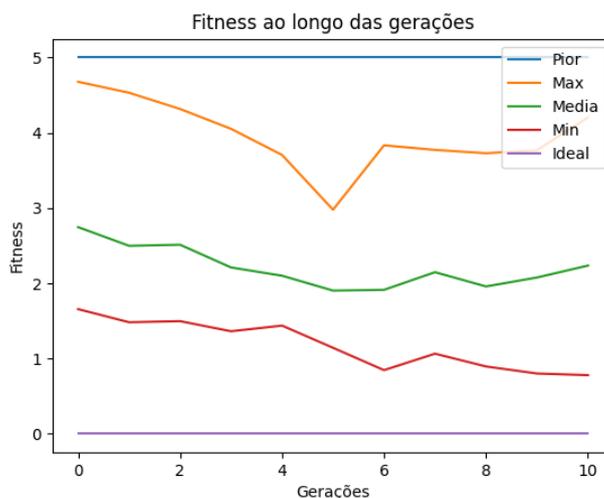
Portanto, um número alto de gerações pode levar a um *fitness* baixo, e é isso o que se espera. Porém, com um número de gerações alto, há a possibilidade de que um valor baixo apareça e acabe se perdendo em uma nova avaliação.

Dentre as 3.250 variações do hiperparâmetros, 3103 tiveram resultados melhores de *fitness* em uma geração antes da final do que na sua geração final. Dentre estes resultados menores, 18 foram menores do que o menor valor reportado na Tabela 5.1, e estão listados abaixo na Tabela 5.2.

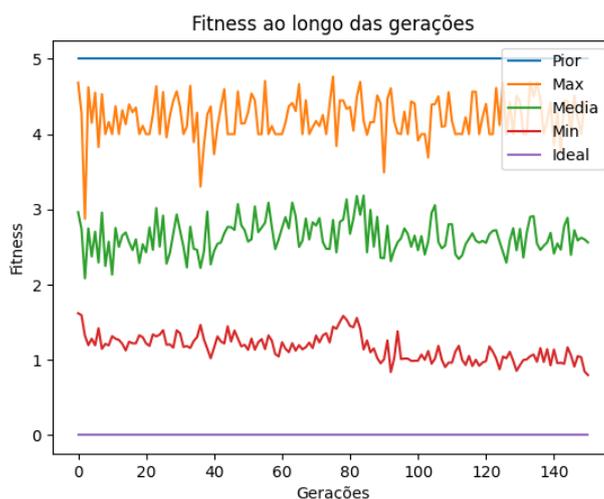
Com estes valores, é possível perceber que eventualmente um maior número de gerações gera *fitness* mais baixos do que execuções com menos gerações. Porém, como comentado anteriormente, se esse valor mais baixo é encontrado muito cedo, ele sofre variações em avaliações futuras. O conjunto de regras é passado adiante para a nova geração, ou seja, seus genes são repassados, porém, podem se perder quando testados novamente.

Existem algumas maneiras de contornar essa situação, como mudar a condição de parada. Ao invés de utilizar o número de gerações, é possível usar a proximidade com o *fitness* ideal. Porém, é difícil quantificar uma diferença aceitável do *fitness* ideal de 0. Em outras palavras, o *fitness* ideal pode ser inatingível e o menor *fitness* possível é desconhe-

Figura 5.2 – *Fitness* ao longo da execução que resultou o melhor *fitness* e o segundo melhor *fitness* ao final do processo.



(a) Configuração 1: 10 gerações, 50 indivíduos na população, 22 indivíduos no torneio, 0,6 de taxa de mutação



(b) Configuração 2: 150 gerações, 30 indivíduos na população, 12 indivíduos no torneio, 0,1 de taxa de mutação

Fonte: A Autora

cido. Outro modo é modificar a etapa de testagem dos jogos. Quando o jogo é testado, são rodadas 100 partidas e todas as métricas são calculadas como uma média destas 100 partidas. Aumentar o número de partidas durante estes testes pode trazer uma avaliação mais precisa, diminuindo a instabilidade. Também, não testar o jogo novamente após o selecionar para o elitismo faria com que ele permanecesse com o *fitness* encontrado. Contudo, existe a possibilidade de esse indivíduo não ter o seu melhor *fitness* neste determinado momento, tendo que ficar com um valor "ruim" durante o resto da execução, além de ficar desatualizado com o restante da população.

Tabela 5.2 – *Fitness* encontrados em gerações não finais cujo valor é menor do que o menor valor encontrado anteriormente entre todas as gerações finais.

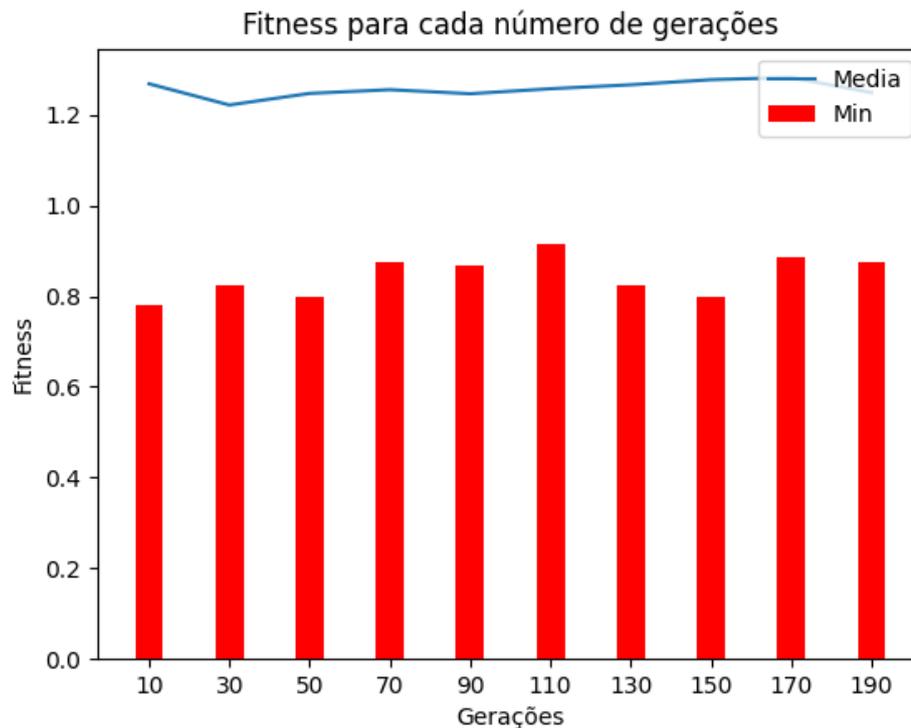
Melhor <i>fitness</i>	Geração	<i>Fitness</i> final	Nº de gerações	Tamanho da população	Tamanho do torneio	Taxa de mutação
0,73037	66	1,13161	190	50	20	0,8
0,74421	21	1,35271	190	50	24	0,2
0,74703	44	1,30192	190	45	14	0,4
0,75092	169	1,16782	170	30	14	0,1
0,75344	100	0,95287	150	40	20	0,1
0,75983	52	1,21769	70	40	14	0,4
0,76543	48	1,11187	170	40	14	0,4
0,76591	30	1,26375	170	45	16	0,6
0,76596	53	1,16783	150	35	2	0,4
0,76607	65	0,94762	70	50	12	0,6
0,76668	32	0,96564	190	45	14	0,8
0,76734	57	1,12516	130	45	20	0,6
0,76929	39	1,23103	50	50	6	0,4
0,77057	80	1,25590	130	35	14	0,6
0,77212	8	0,98974	190	40	16	0,8
0,77369	93	1,00685	170	20	10	0,8
0,77512	63	1,08843	110	50	22	0,6
0,77794	42	1,05435	90	50	20	0,1

Fonte: A Autora

Seguindo com as avaliações dos hiperparâmetros, podemos extrair os melhores valores para cada um independente dos outros e também entender como cada característica influencia no resultado do *fitness*. Analisando primeiro o número de gerações, apresentado na Figura 5.3, podemos ver que não existe um padrão bem definido.

Os valores mostrados nos gráficos são a média de cada conjunto de execuções e o menor valor encontrado na última geração do conjunto de execuções. O *fitness* mínimo começa baixo, aumenta e depois diminui com as execuções de 150 gerações. Já o valor médio se mantém sem grande variação. Com isso, podemos entender que o número de gerações avaliado sozinho não tem grande influência no resultado do *fitness*, uma vez que não foi possível encontrar um padrão significativo.

Passando para o número de indivíduos na população, mostrado na Figura 5.4, foi possível observar que há efeito em ambos os valores mínimos e médios. Quanto maior o número de indivíduos na população, menor é a média do *fitness*. Ou seja, esta característica pode não influenciar drasticamente no valor mínimo, mas tem a sua parcela de responsabilidade no valor médio. Com 50 indivíduos em uma população, por exemplo,

Figura 5.3 – Variação do *fitness* ao longo dos valores de geração.

Fonte: A Autora

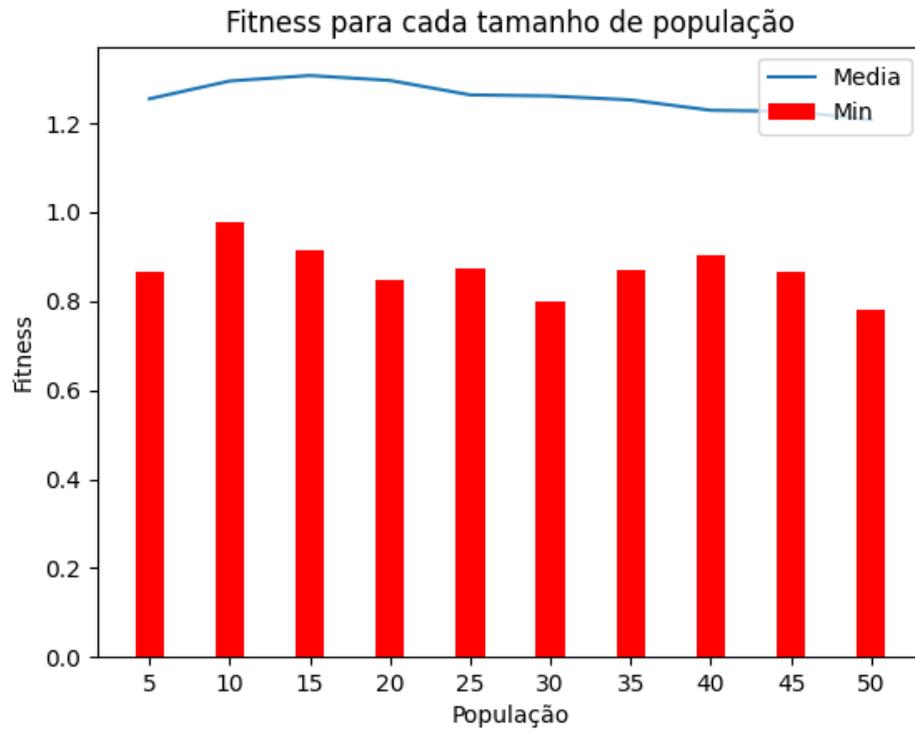
o nível de diversidade é muito elevado, sendo possível cobrir cerca de 15% do total de combinações iniciais de mapas disponíveis e demais parâmetros.

Quando analisamos o número de indivíduos no torneio, apresentado na Figura 5.5, vemos que o gráfico segue o mesmo padrão do tamanho da população. Uma variação nos valores de *fitness* mínimo e um decréscimo na média conforme o tamanho do torneio aumenta. Com isso, o que se pode concluir deste hiperparâmetro é que ele não serve como medida para encontrar o menor *fitness*, mas sim para diminuir os valores em um modo geral. Porém, como foi visto na Tabela 5.1, os melhores *fitness* tiveram valores de tamanho de torneio bem variados, porém com prevalência dos menores.

Por fim, para completar a análise dos hiperparâmetros, a taxa de mutação segue os mesmos padrões vistos anteriormente, conforme a Figura 5.6. O valor mínimo de *fitness* sofre uma queda na medida que a taxa de mutação aumenta, mas também cresce no último valor. Já a média dos *fitness* de cada taxa segue em queda até o maior valor, mantendo o padrão dos demais hiperparâmetros. Como a quantidade de valores é menor do que os outros, não é possível perceber uma alternância tão grande.

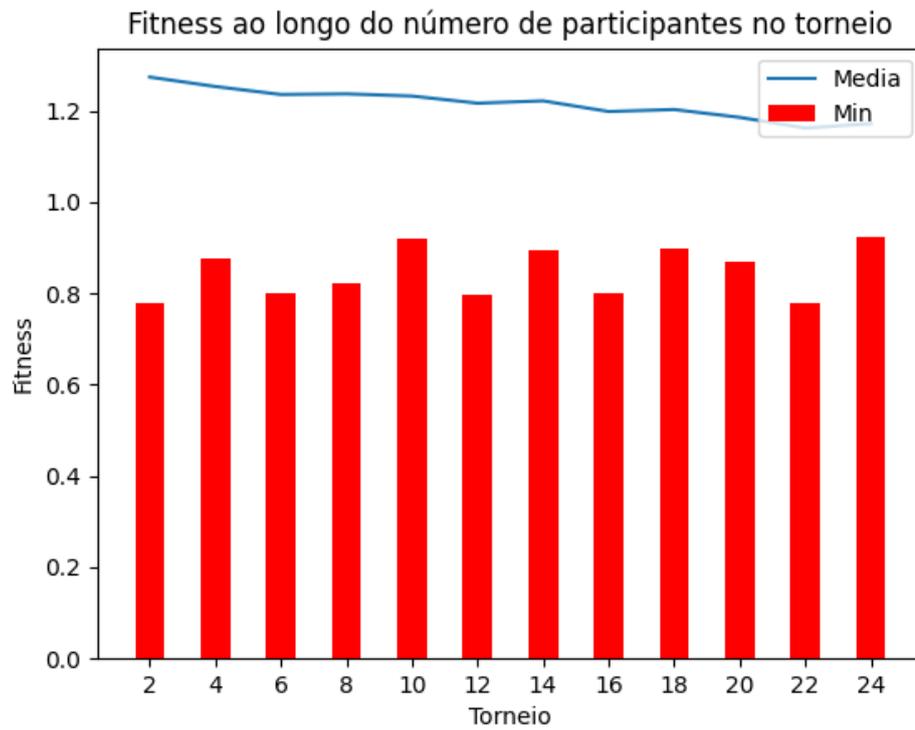
Em síntese, não há um hiperparâmetro com total responsabilidade sobre os valores finais de *fitness*. O que há é apenas uma diminuição dos valores médios, fazendo com que

Figura 5.4 – Variação do *fitness* ao longo dos valores de tamanho da população.



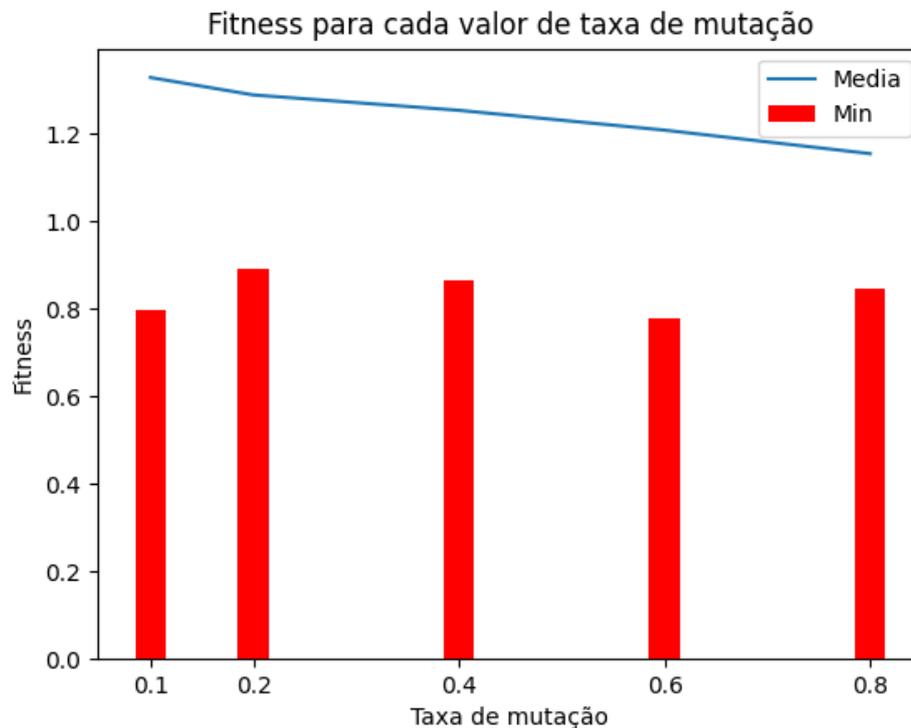
Fonte: A Autora

Figura 5.5 – Valores de *fitness* ao longo dos valores de tamanho do torneio.



Fonte: A Autora

Figura 5.6 – Valores de *fitness* ao longo dos valores de taxa de mutação.



Fonte: A Autora

toda a população seja melhor de uma forma geral. Porém, estes valores não chegam tão próximos do valor ideal ou mesmo do mínimo. De qualquer modo, as execuções utilizadas nos próximos testes estarão entre as 10 melhores mostradas na Tabela 5.1.

5.2 Evolução do Mapa

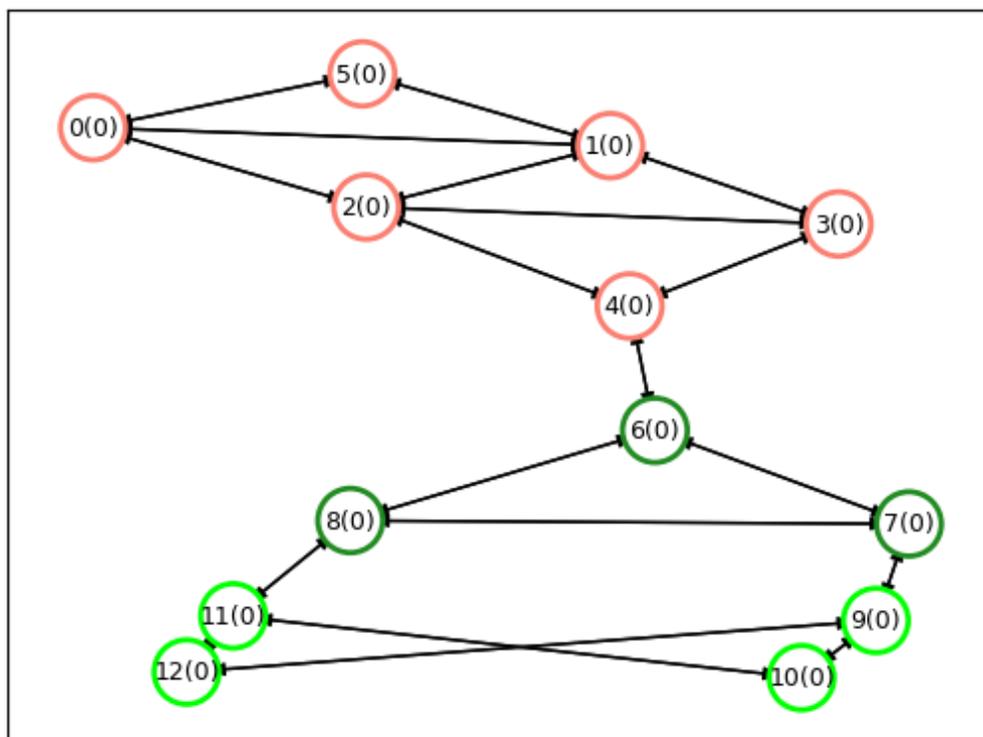
A presente seção se dedica a avaliar um dos aspectos mais importantes do jogo: o mapa. Ele tem o poder de influenciar todas as métricas de avaliação. O fator de ramificação e a duração são os mais atingidos, uma vez que os movimentos possíveis consideram a quantidade de territórios no mapa e quanto mais territórios, mais tempo leva para conquistar o mapa inteiro. Ao longo desta seção são avaliados o isomorfismo dos mapas e a viabilidade dos mesmos. O isomorfismo foi analisado com o método *is_isomorphic*¹ da biblioteca Networkx. O método utilizado para essa verificação é definido por Foggia, Sansone and Vento (2001)

O primeiro passo é identificar mapas com grafos isomorfos, com o objetivo de

¹https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.isomorphism.is_isomorphic.html

verificar se houve populações com predominância de um mapa. Duas execuções tiveram essa característica, ou seja, todos os mapas da última geração de ambas as execuções são isomorfos de uma mapa inicial, sendo ambas executaram com 10 gerações e 10% de taxa de mutação. Estas duas características podem ter contribuído fortemente para a propagação de somente um mapa, o mapa 9, mostrado na Figura 5.7. Da população inicial, somente 2 indivíduos carregavam este mapa. Na população final, todos os indivíduos têm mapas isomorfos em relação ao mapa 9. Isso aconteceu, provavelmente, devido ao baixo número de territórios deste mapa, o que gerou valores baixos dos critérios de fator de ramificação e duração, melhorando o resultado do *fitness*.

Figura 5.7 – Representação do mapa inicial 9.



Fonte: A Autora

Ao todo, 65 combinações de mapas isomorfos foram encontradas, totalizando 30 indivíduos da população final de uma execução e 35 indivíduos de uma segunda execução. No entanto, são apenas duas configurações de parâmetros de um total de 3250 combinações avaliadas. Isso não faz com que o processo de evolução seja considerado ruim, visto que o acontecimento não foi recorrente.

O principal ponto forte dos mapas 9 é a pouca quantidade de territórios, se comparado com os mapas reais de Risk. Com 13 territórios, o tempo de jogo se reduz drasticamente, assim como o fator de ramificação. Por mais que não afete drasticamente os

demais critérios, o decréscimo no número de movimentos possíveis é algo que nem todos os mapas conseguem oferecer.

Em média, os melhores indivíduos de todas as execuções de teste tem o mínimo de 2 territórios, máxima de 24 e média de 3,39 territórios. Mesmo o indivíduo com maior número de territórios tem menos do que os mapas reais de Risk, que possuem 42 territórios.

Em outra análise, uma comparação dos mapas da execução mais rápida com os mapas da execução mais longa foi feita para entender os efeitos dos pontos extremos dos hiperparâmetros. Os hiperparâmetros destas duas execuções podem ser visto na Tabela 5.3.

Tabela 5.3 – Parâmetros das execuções mais simples e mais exaustiva.

Execução	Número de gerações	Número de população	Tamanho do torneio	Taxa de mutação
Rápida	10	5	2	0,1
Longa	190	50	24	0,8

Fonte: A Autora

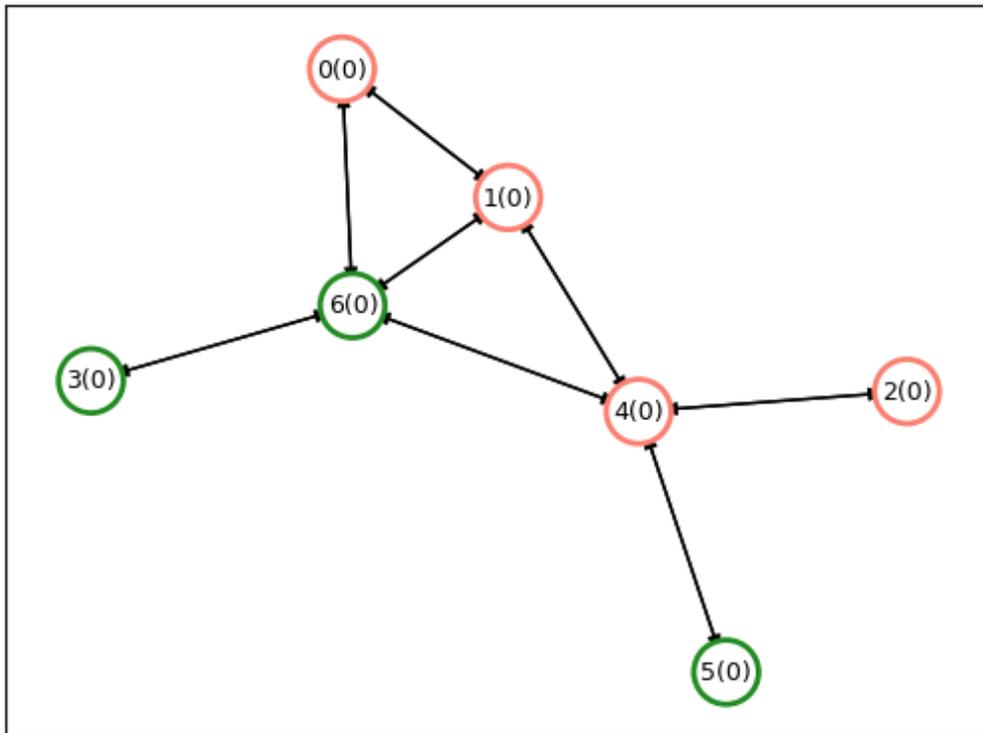
O mapa do melhor indivíduo gerado pela execução mais rápida pode ser visto na Figura 5.8 e da execução mais longa está na Figura 5.9. Ambos os mapas possuem uma quantidade bem reduzida de territórios, sendo que o mapa da execução rápida possui 7 território e o da longa possui 2 territórios, muito próximo da média.

Olhando para os critérios, mostrados na Tabela 5.4, podemos ver porque os melhores indivíduos tiveram esses mapas. A vantagem dos dois fica muito perto do valor ideal, sendo que a execução simples conseguiu atingir este valor ideal, e o fator de ramificação também permaneceu perto do valor ideal. Além disso, a duração também se manteve reduzida em ambas as execuções.

Os valores considerados ideais para os critérios tem uma limitação. Os critérios de duração e fator de ramificação favorecem o surgimento de mapas pequenos, como o da Figura 5.9, pois o número de territórios tem muita influência nesses critérios. Então, como a função de *fitness* exige que esses critérios tenham valor baixo, o número de territórios precisa ser pequeno. Em mapas como o da Figura 5.9, a vitória do primeiro jogador é praticamente certa, o que é ruim, mesmo que ele tenha satisfeito a função de *fitness* definida.

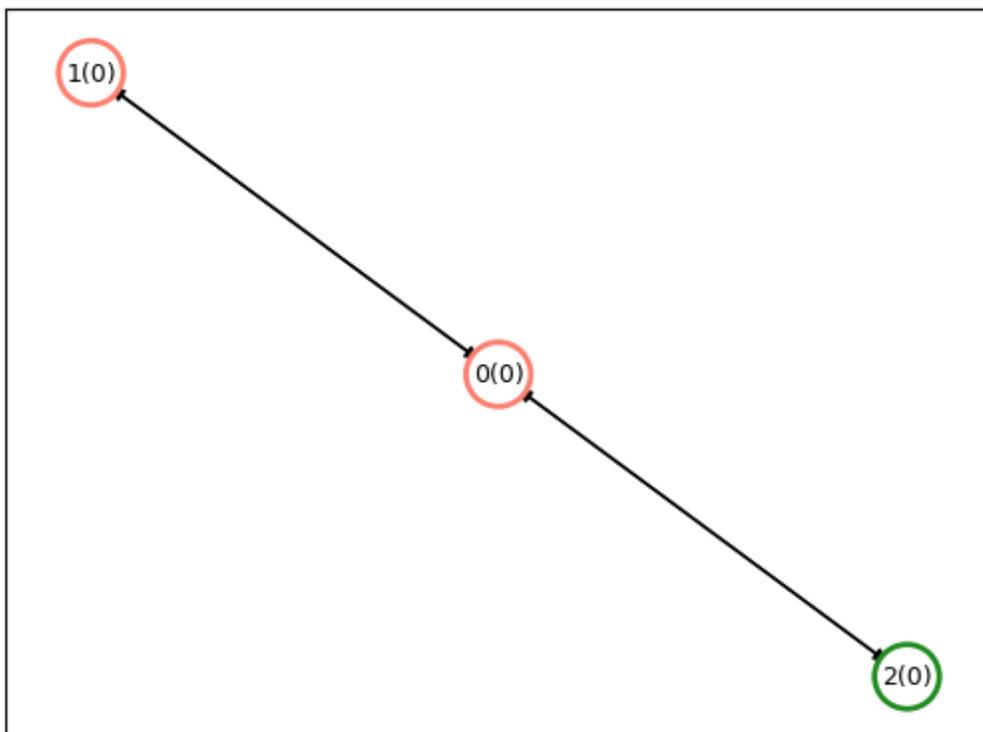
Dentre todas as 3250 execuções, 3236 (99,56%) tem o seu melhor indivíduo com

Figura 5.8 – Representação do mapa do melhor indivíduo da execução rápida.



Fonte: A Autora

Figura 5.9 – Representação do mapa do melhor indivíduo da execução longa.



Fonte: A Autora

Tabela 5.4 – Critérios das execuções mais rápida e mais longa.

Critérios	Critérios Ideal	Critérios Rápida	Distância Rápida	Critérios Longa	Distância Longa
Compleitude	1	0,98	0,02	1	0
Duração	0	0,41	0,41	0,69	0,69
Vantagem	0	0	0	0,02	0,02
Fator de Ramificação	0,5	0,76	0,26	0,50	0,
Drama	0,5	0,30	0,2	0,42	0,08
Movimentos Extremos	0,5	0,78	0,28	0,56	0,06
Mudança de Liderança	0,5	0,25	0,25	0,42	0,08
<i>Fitness</i>	0		1,42		0,93

Fonte: A Autora

até 10 territórios. Isso mostra uma grande predominância de mapas pequenos, sendo possível concluir que, para alcançar os valores ideais dos critérios, os mapas gerados precisam ser pequenos.

5.3 Evolução dos Jogos

Após entender a evolução dos mapas, precisamos verificar também o restante dos parâmetros. Apesar de possuírem um conjunto menor de combinações, eles são tão importantes quanto o mapa no cumprimento dos objetivos deste trabalho.

No geral, os parâmetros não possuem predominância na geração final. Os dois valores do parâmetro de modo de distribuição inicial de territórios ficou muito bem dividido:

- Territórios distribuídos aleatoriamente: 1900
- Jogadores escolhem os territórios: 1350

Estes valores sugerem que este parâmetro tem pouca influência sobre o desempenho do jogo. Quando os jogadores escolhem seus territórios, há uma grande chance de ambos começarem a partida com a vantagem de possuir um continente inteiro. Quando a distribuição é aleatória, nenhum dos dois tem essa vantagem. Assim, nos dois modos da distribuição dos territórios, a partida fica igualmente equilibrada, mas com uma tendência para a distribuição aleatória.

Já observando o parâmetro de quantidade de dados defensivos, há uma certa diferença na quantidade de jogos com cada valor, porém não é muito significativa:

- 2 dados: 1674
- 3 dados: 1576

Quando somente 2 dados defensivos são utilizados, há uma vantagem para o lado atacante, visto que o jogador tem uma chance a mais de conseguir um número maior do que o oponente em seus dados. Já com 3 dados, essa chance diminui, uma vez que a quantidade de dados de cada jogador é a mesma. Com isso, podemos ver uma pequena tendência para o favorecimento ao atacante.

No parâmetro de quantidade de tropas movidas na conquista de novo território, é possível observar uma diferença maior:

- Quantidade mínima (número de dados atacantes - 1): 2629
- Quantidade máxima (número de tropas no território atacante - 1): 621

Quando a quantidade mínima é transferida para o novo território, o número de possíveis ataques a partir deste território conquistado são pequenas e provavelmente limitam a atuação em sequência do jogador. Já com a quantidade máxima, a partida pode se tornar mais longa. Este parâmetro impacta consideravelmente na duração da partida, uma vez que indica o quão longe o jogador pode ir.

Por último, o fator que calcula a quantidade de tropas recebidas no início de cada turno obteve resultados mais uniformes:

- Fator = 1: 773
- Fator = 2: 807
- Fator = 3: 845
- Fator = 4: 825

Considerando o número mínimo de tropas que devem ser recebidas pelo jogador no início de seu turno, que são 3, é preciso ter 12 territórios para obter essa quantidade. Como comentado anteriormente na Seção 5.2, o número médio de territórios nos mapas dos indivíduos da última geração das execuções de teste foi de 3,39. Com isso, se impossibilita conseguir mais do que 3 tropas, tornando o fator irrelevante.

A partir destes resultados, podemos concluir que o melhor indivíduo teria os territórios distribuídos aleatoriamente, 3 dados para o defensor, uma vez que existe pouca diferença com a utilização de 2 dados, quantidade mínima de tropas movidas na conquista de um território e fator igual a 3. O indivíduo com o melhor *fitness* tem exatamente estes parâmetros.

5.4 Jogabilidade, Versões Geradas e Versões Reais

Testar a jogabilidade dos melhores jogos gerados requer uma grande quantidade de participantes humanos com responsabilidade e tempo disponível. Por isso, dentro das limitações de tempo, o único teste real feito foi pela autora.

Os jogos testados são jogáveis e consistentes com os valores ideais dos critérios de avaliação. Porém, somente com testes de jogadores humanos será possível avaliar se os jogos gerados são divertidos, desafiadores e engajantes o suficiente.

Considerando que o fator que mais pode variar no processo de evolução é o tamanho do mapa, jogos das gerações iniciais possuem mapas maiores do que os gerados no final. Porém, somente este critério não é suficiente para concluir se um jogo é bom ou ruim, pois pode haver uma combinação dos demais parâmetros que torne bom o jogo em um mapa considerado ruim.

A avaliação das versões de Risk e War, no entanto, podem ser comparadas com a dos jogos gerados. Os parâmetros do Risk incluem livre escolha inicial de territórios, fator igual a 3 no cálculo de tropas recebidas, 2 dados de defesa e número máximo de tropas movidas na conquista de um novo território. Os parâmetros do War incluem designação por cartas de territórios iniciais, fator igual a 2 no cálculo de tropas recebidas, 3 dados de defesa e número mínimo de tropas movidas na conquista de um novo território. Além disso, é importante lembrar que o número máximo de turnos jogados antes de declarar a partida como não terminada é de 24, sendo 12 para cada jogador. Os valores dos critérios de War e Risk, além dos critérios ideais e do melhor jogo gerado, se encontram na Tabela 5.5

Tabela 5.5 – Valores dos critérios calculados para os parâmetros originais de Risk e War

Critérios	Risk	War	Ideais	Melhor gerado
Compleitude	0.06333	0,99666	1	1,0
Drama	0.89075	0,11119	0,5	0,42748
Duração	0.99499	0,33194	0	0,63708
Fator de Ramificação	1	1	0,5	0,51499
Movimentos Extremos	0.27380	0,47390	0,5	0,47584
Mudança de Liderança	0.00988	0,12070	0,5	0,48917
Vantagem	0.96	0,68	0	0,02
<i>Fitness</i>	4.49873	2,30949	0	0,77956

Fonte: A Autora

O fator de ramificação de ambos foi de 1, mostrando que existem muitas jogadas possíveis em cada turno. No cálculo do fator de ramificação, as possibilidades são truncadas em 100. Então, se o fator de ramificação for igual a 1, significa que o jogador teve uma média de 100 possíveis movimentações na rodada de testes. Também, a vantagem de ambos foi alta, 0,96 no Risk e 0,68 no War, significando que o primeiro jogador possui mais chance de ganhar do que o segundo. A completude no Risk foi extremamente baixa, expondo o lado extenso de uma partida, chegando no limite de turnos imposto, que é igual a 24. Já no War, isso não se repetiu, possuindo um valor próximo de 1.

A duração do Risk possui um valor muito alto, mostrando que as partidas tem um número muito superior ao considerado ideal. A duração do War, por outro lado, se mostrou menor, mas ainda acima do ideal. Também, o Risk possui um equilíbrio de vantagem, o que não se repetiu no War. As demais métricas ajudam a estabilizar a avaliação e representam grande parte da diversão em uma partida. Os parâmetros originais de Risk e War resultam em versões diferentes de um mesmo jogo e com métricas bem distintas.

É importante ressaltar que os parâmetros são aplicados a uma implementação que simplifica as regras do jogo e são jogados com um agente baseado em regras. Os parâmetros originais nas regras originais são adequados para jogadores humanos, dado o sucesso comercial dos jogos e a popularidade entre os jogadores.

5.5 De Risk a War

Por fim, esta seção de resultados tem por objetivo testar a capacidade do *pipeline*. Uma vez que os critérios ideias propostos não foram atingidos, foi feita uma tentativa de gerar um jogo já existente e com critérios mais acessíveis. O sistema da Ludi conseguiu criar um jogo completamente novo, convencendo pessoas de que o jogo era criado por outra pessoa e não por um computador, conforme relatado em Browne (2011). O desafio proposto aqui foi: o *pipeline* desenvolvido consegue sair dos parâmetros do Risk e chegar nos do War? O computador poderia (re)criar o War a partir do Risk?

Em todas as execuções testadas nos testes apresentados na Seção 5.1, não há um conjunto de parâmetros que coincide com os do War, considerando a semelhança de mapa também. O desafio é particularmente difícil, pois poderiam haver 10 diferentes mapas na população inicial. Porém, pelo número de execuções testadas, imaginava-se que seria suficiente para gerar pelo menos um mapa que fosse igual ao do War. Porém, isso não

ocorreu.

Contudo, um novo teste foi feito. Dessa vez, a população inicial foi composta somente de jogos Risk, com mapa e parâmetros iguais e reais, dentro das condições da implementação simplificada. Foram utilizadas as 10 melhores configurações de hiperparâmetros apresentadas na Tabela 5.1.

Para medirmos a dificuldade na evolução do mapa, o grafo do mapa do Risk básico se encontra na Figura 2.1 e o grafo do mapa do War básico, na Figura 2.2. Eles têm muitas semelhanças, mas algumas fronteiras são diferentes.

No mapa do Risk, Ontario tem ligação com Groelândia, mas no mapa do War, Ottawa, que representaria a região de Ontario, não tem ligação com a Groelândia. A Austrália, no War, passa a ser dividida em Western Australia e Eastern Australia, além da subtração de Bornéu. Entre outras modificações, estão também as nomenclaturas das regiões e a divisão da Europa. No War, são utilizados os nomes dos países, como França, Alemanha e Polônia. Já no Risk, são representados como Western, Northern e Southern Europe.

Mesmo com a população inicial ajustada para haver predominância de mapas de Risk, o mapa do War não foi encontrado a partir dele. Porém, o restante dos parâmetros foi encontrado 2081 vezes simultaneamente no indivíduo, de um total de 29.200 indivíduos gerados, conforme especificado na Tabela 5.6, mostrando que a maior dificuldade é o mapa.

Tabela 5.6 – Número de indivíduos gerados com os parâmetros do War nas 10 melhores configurações.

Execução	Nº de indivíduos com parâmetros do War	Nº de indivíduos gerados	Taxa de sucesso
1	17	500	3,4%
2	122	4500	2,7%
3	187	2500	7,48%
4	287	4500	6,37%
5	588	3000	19,6%
6	228	7500	3,04%
7	55	1350	4,07%
8	349	4050	8,61%
9	197	1050	18,76%
10	51	250	20%

Fonte: A Autora

Conforme os parâmetros da Tabela 5.1, as melhores taxas de sucesso são de con-

figurações com altos valores de taxa de mutação. Provavelmente, a alta mutabilidade favorece a aparição de parâmetros que não eram muito utilizados e a combinação dos parâmetros do War não é a única que gera partidas com os critérios calculados na Tabela 5.5. Da mesma forma, os conjuntos de parâmetros encontrados pelo GA tiveram limitações, como os mapas pequenos, e os parâmetros de ambos Risk e War são apropriados para a versão completa, não para a versão simplificada.

5.6 Resumo

O primeiro experimento feito foi a testagem dos hiperparâmetros e quais combinações são capazes de gerar indivíduos com melhor *fitness*. Os resultados são apresentados na Tabela 5.1. A partir das Figuras 5.3 a 5.6 foi possível ver que o aumento nos valores dos hiperparâmetros diminuiu somente a média do *fitness* ao longo das gerações.

Na análise da evolução dos mapas, percebemos que a prevalência é de mapas menores, onde 99,56% dos melhores indivíduos de todas as execuções possuem 10 territórios ou menos. Essa característica se deve, provavelmente, aos valores das métricas de fator de ramificação e distância, uma vez que para ter menos movimentos possíveis e jogos mais curtos é necessário diminuir a quantidade de territórios.

Além disso, uma análise sobre a evolução dos demais parâmetros nos mostra que existem valores que favorecem a aparição de jogos melhores. No modo de distribuição dos territórios, o que mais prevaleceu nos melhores indivíduos foi a distribuição aleatória, assim como 2 dados para o defensor, a quantidade mínima de tropas movidas na conquista de um território e fator igual a 3. Também, estes são os parâmetros do indivíduo com melhor *fitness*.

Por fim, os parâmetros do Risk e War foram testados na versão simplificada do Risk apresentada neste trabalho. Os dois jogos tiveram *fitness* altos se comparado com a melhor versão gerada pelo GA. Também, um teste para gerar o conjunto de regras do War a partir das regras do Risk foi feito, porém, sem sucesso.

5.7 Análise

Levando em consideração os experimentos e resultados apresentados, é possível concluir que o pipeline consegue gerar novas variações do Risk. Todas as variações gera-

das são jogáveis, porém as que passaram por mais gerações ficaram com mapas pequenos, com a média de 3,39 territórios. Estas versões resultam em partidas com no máximo 72 minutos, como mencionado na Seção 4.3.5.2. Porém, além de diminuir a duração, outro objetivo codificado na função de *fitness* era diminuir o fator de ramificação, a fim de encontrar versões mais simples do jogo. Isso também favorece a redução do número de territórios no mapa. Com mapas com 4 territórios, uma partida não passa de meia hora, considerando que a quantidade de ataques possíveis passa a ser no máximo 3. Portanto, a melhora no fator de ramificação pode precisar de outros parâmetros ou regras além do escopo deste trabalho, como limitar a quantidade mínima de tropas para atacar.

Uma limitação do trabalho é que o processo de evolução depende fortemente do agente utilizado para o jogo. É preciso implementar um agente bom para gerar versões também boas. Os jogos gerados serão medidos pelas ações dos jogadores e se eles jogarem de uma forma básica, os jogos também serão básicos. O jogador baseado em regras (Seção 4.3.4.1) é rápido, porém básico. Por outro lado, algoritmos de IA que conseguem bom desempenho em jogos de tabuleiro não são generalistas.

A utilização de algoritmo genético traz uma nova versão dos experimentos que antes eram feitos com programação genética. Com o algoritmo genético, a exploração dos parâmetros é mais agressiva e pode trazer mais variabilidade aos indivíduos gerados.

6 CONCLUSÃO

Este trabalho apresentou a extensão do pipeline de design evolutivo de jogos para jogos de mesa no estilo Risk e War. A versão do jogo é representada por um conjunto de parâmetros relacionados às regras que sofrem variações. O GA descrito ao longo deste trabalho é usado para encontrar a melhor combinação possível destes parâmetros. A função de *fitness* é calculada pela soma das distâncias das métricas. Ao final, as versões geradas são testadas com um agente baseado em regras.

A exploração do jogo Risk, pouco feita até o momento, foi realizada de diversas formas durante a execução deste trabalho. Desde a extensão de um agente baseado em regras até a criação de uma nova versão de Risk, o jogo serviu de base para o teste de diversos métodos em diferentes áreas. Foi possível entender o *crossover* de grafos e suas principais dificuldades na execução da integração, além das várias formas de analisar a qualidade de jogos.

Trabalhos futuros podem abordar as limitações identificadas no trabalho. A primeira é a instabilidade do *fitness* devido à aleatoriedade do jogo. Um maior número de partidas para calcular as métricas pode deixar os valores mais estáveis.

Além disso, um estudo mais aprofundado sobre as métricas que compõe o *fitness* e seus valores-alvo pode aumentar a qualidade dos jogos gerados. Também, a geração procedural de mapas pode favorecer a variabilidade da população inicial, trazendo mapas mais diversos.

Também, o desenvolvimento de um agente mais sofisticado é de extrema importância. O desafio é que o agente deve lidar com a complexidade do jogo de maneira muito eficiente, já que uma simples avaliação de *fitness* requer centenas de partidas. Alguma extensão de MCTS (BROWNE et al., 2012) ou RMEA (ISLAM; ALAM; MURASE, 2007), utilizados em jogos digitais, pode ser promissora.

Ainda, testes com jogadores humanos, podem apresentar outras perspectivas e olhares sobre o jogo e suas versões. No futuro, a continuação deste trabalho necessita de grupos de teste para encontrar versões melhores e, até mesmo, na definição de valores ideais para as métricas de avaliação.

Por fim, ajustar este trabalho para os mais variados jogos é um projeto interessante. Agregar desde cartas até peças de diferentes movimentações são desafios que podem ser criados. Um desafio ainda maior é o estabelecimento de uma linguagem de descrição de jogos para jogos de mesa, possibilitando a criação de jogos completamente inéditos.

REFERÊNCIAS

- AARSETH, E. From literature to cultural literacy. In: SEGAL, N.; KOLEVA, D. (Ed.). Palgrave Macmillan UK, 2014. chp. I Fought the Law: Transgressive Play and the Implied Player, p. 180–188. ISBN 978-1-137-42970-4. Disponível em: <https://doi.org/10.1057/9781137429704_13>.
- ARAÚJO, B. de A. S. et al. O ensino de geopolítica através da produção textual e do jogo war: Práticas docentes do PIBID Geografia-UFRN. **Revista Extensão & Sociedade**, Universidade Federal do Rio Grande do Norte - UFRN, v. 9, 5 2018. ISSN 2178-6054. Disponível em: <<https://periodicos.ufrn.br/extensoesociedade/article/view/14487>>.
- ASHLOCK, D. Automatic generation of game elements via evolution. **Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games, CIG2010**, p. 289–296, 2010.
- BOMBARDELLI, L. B. **Generating variations of the board game risk through evolutionary game design**. Trabalho de Conclusão de Curso — Universidade Federal do Rio Grande do Sul, 2022. Disponível em: <<https://lume.ufrgs.br/handle/10183/252514>>.
- BORVO, A.; SEGUIN, J. **Anatomie d'un jeu de cartes: L'aluette ou le jeu de la vache**. FeniXX réédition numérique, 1977. 106 p. ISBN 9782402230445. Disponível em: <<https://books.google.com.br/books?id=h9xYDwAAQBAJ>>.
- BOYER, J. M. et al. Stop minding your P's and Q's: Implementing a fast and simple DFS-based planarity testing and embedding algorithm. In: SPRINGER. **International Symposium on Graph Drawing**. [S.l.], 2003. p. 25–36.
- BRAND, D.; KROON, S. Sample evaluation for action selection in Monte Carlo tree search. In: **Proceedings of the Southern African Institute for Computer Scientist and Information Technologists Annual Conference 2014 on SAICSIT 2014 Empowered by Technology**. Association for Computing Machinery, 2014. p. 314–322. ISBN 9781450332460. Disponível em: <<https://dl.acm.org/doi/10.1145/2664591.2664612>>.
- BROWNE, C. **Automatic Generation and Evaluation of Recombination Games**. Thesis (PhD) — Queensland University of Technology, 2008. Disponível em: <https://eprints.qut.edu.au/17025/1/Cameron_Browne_Thesis.pdf>.
- BROWNE, C. **Evolutionary Game Design**. Springer London, 2011. (SpringerBriefs in Computer Science). ISBN 9781447121794. Disponível em: <<https://doi.org/10.1007/978-1-4471-2179-4>>.
- BROWNE, C. B. et al. A survey of monte carlo tree search methods. **IEEE Transactions on Computational Intelligence and AI in games**, IEEE, v. 4, n. 1, p. 1–43, 2012.
- CARR, J. Using graph convolutional networks and TD(λ) to play the game of risk. 2020. Disponível em: <<https://arxiv.org/abs/2009.06355>>.
- CHEN, T.; GUY, S. Chaos cards: Creating novel digital card games through grammatical content generation and meta-based card evaluation. **Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment**, v. 16, n. 1, p. 196–202, 2020. Disponível em: <<https://ojs.aaai.org/index.php/AIIDE/article/view/7430>>.

DARWIN, C. **On the Origin of Species by Means of Natural Selection or the Preservation of Favored Races in the Struggle for Life**. London: Murray, 1859.

DEMPSEY, J. V. et al. Forty simple computer games and what they could mean to educators. **Simulation & Gaming**, v. 33, n. 2, p. 157 – 168, 2002. Disponível em: <<https://doi.org/10.1177/1046878102332003>>.

FERRARI, R.; ASSUNÇÃO, J. Towards playing risk with a hybrid Monte Carlo based agent. In: **Anais Estendidos do XXI Simpósio Brasileiro de Jogos e Entretenimento Digital**. Porto Alegre, RS, Brasil: SBC, 2022. p. 301–306. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/sbgames_estendido/article/view/23664>.

FOGGIA, P.; SANSONE, C.; VENTO, M. An improved algorithm for matching large graphs. In: . [S.l.: s.n.], 2001.

GAINA, R. D. et al. TAG: A Tabletop Games Framework. In: **Experimental AI in Games (EXAG), AIIDE 2020 Workshop**. [S.l.: s.n.], 2020.

GENESERETH, M.; LOVE, N.; PELL, B. General game playing: Overview of the AAAI competition. **AI Magazine**, v. 26, n. 2, p. 62–72, 06 2005. Disponível em: <<https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/1813>>.

GIBSON, R.; DESAI, N.; ZHAO, R. An automated technique for drafting territories in the board game Risk. **Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment**, v. 6, n. 1, p. 15–20, 2010.

HARJU, M. On probabilities of Risk type board game combats. 2012. Disponível em: <<https://arxiv.org/abs/1204.4082v1>>.

HENDEL, S. et al. Taking the risk out of RISK: Conquer odds in the board game RISK. p. 1–14, 2015. Disponível em: <<https://arxiv.org/abs/1512.04333v1>>.

HENDRIKX, M. et al. Procedural content generation for games: A survey. **ACM Trans. Multimedia Comput. Commun. Appl.**, Association for Computing Machinery, New York, NY, USA, v. 9, n. 1, feb 2013. ISSN 1551-6857. Disponível em: <<https://doi.org/10.1145/2422956.2422957>>.

HEREDIA, L. G.; CAZENAVE, T. Expert iteration for Risk. In: **Advances in Computer Games: 17th International Conference, ACG 2021, Virtual Event, November 23–25, 2021, Revised Selected Papers**. Springer-Verlag, 2021. p. 27–37. ISBN 978-3-031-11487-8. Disponível em: <https://doi.org/10.1007/978-3-031-11488-5_3>.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. The MIT Press, 1975. 232 p. ISSN 9780262581110. Disponível em: <<https://mitpress.mit.edu/9780262581110/adaptation-in-natural-and-artificial-systems/>>.

HOM, V.; MARKS, J. Automatic design of balanced board games. **Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment**, v. 3, p. 25–30, 2007. ISSN 2334-0924. Disponível em: <<https://ojs.aaai.org/index.php/AIIDE/article/view/18777>>.

ISLAM, M. M.; ALAM, M. S.; MURASE, K. A new recurring multistage evolutionary algorithm for solving problems efficiently. In: YIN, H. et al. (Ed.). **Intelligent Data Engineering and Automated Learning - IDEAL 2007**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 97–106. ISBN 978-3-540-77226-2.

KELLEY, D. **Art of Reasoning: An Introduction to Logic and Critical Thinking**. W. Norton, Incorporated, 2013. ISBN 9780393740080. Disponível em: <<https://books.google.com.br/books?id=50xEvQEACAAJ>>.

MARKS, M. P. Using the game of Risk to teach international relations. **International Studies Notes**, Oxford University Press, v. 23, n. 1, p. 11–18, 1998. ISSN 00947768. Disponível em: <<http://www.jstor.org/stable/44235311>>.

MEDEIROS, D. P.; GONÇALVES, M. M. A metáfora do jogo: Relações entre metodologia projetual e jogos de tabuleiro. **XXI Simpósio Nacional de Geometria Descritiva e Desenho Técnico**, 2013.

MITCHELL, M. Genetic algorithms: An overview. **Complexity**, v. 1, p. 31–39, 1995.

NIELSEN, T. S. et al. Towards generating arcade game rules with VGDL. **2015 IEEE Conference on Computational Intelligence and Games (CIG)**, p. 185–192, 2015.

OLSSON, F. **A Multi-Agent System for playing the board game Risk**. Dissertation (Master) — Blekinge Institute of Technology, 2005. Disponível em: <<https://www.diva-portal.org/smash/get/diva2:831093/FULLTEXT01.pdf>>.

PITRAT, J. Realization of a general game-playing program. **IFIP Congress**, p. 1570–2574, 1968.

RISI, S.; PREUSS, M. From Chess and Atari to Starcraft and beyond: How game AI is driving the world of AI. **KI - Kunstliche Intelligenz**, Springer Science and Business Media Deutschland GmbH, v. 34, p. 7–17, 3 2020. ISSN 16101987.

ROEHRS, R. Adaptação do jogo War® como ferramenta didática aplicada ao ensino de geografia - WARGEO. **Geografia, Ensino e Pesquisa**, v. 21, p. 99–107, 2017. ISSN 2236-4994.

SALEN, K.; ZIMMERMAN, E. **Rules of Play: Game Design Fundamentals**. MIT Press, 2003. (ITPro collection). ISBN 9780262240451. Disponível em: <<https://mitpress.mit.edu/9780262240451/rules-of-play/>>.

SIVANANDAM, S. N.; DEEPA, S. N. Introduction to genetic algorithms. **Introduction to Genetic Algorithms**, Springer Berlin Heidelberg, p. 1–442, 2008.

SNODGRASS, S.; ONTAÑÓN, S. Learning to generate video game maps using markov models. **IEEE Transactions on Computational Intelligence and AI in Games**, v. 9, n. 4, p. 410–422, 2017.

SUMMERVILLE, A.; MATEAS, M. Mystical tutor: A magic: The gathering design assistant via denoising sequence-to-sequence learning. **Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment**, v. 12, n. 1, p. 86–92, 2016.

TOGELIUS, J.; SCHMIDHUBER, J. An experiment in automatic game design. **2008 IEEE Symposium on Computational Intelligence and Games, CIG 2008**, p. 111–118, 2008.

TOGELIUS, J. et al. Search-based procedural content generation: A taxonomy and survey. **IEEE Transactions on Computational Intelligence and AI in Games**, v. 3, n. 3, p. 172–186, 2011.

URIARTE, A.; ONTAÑÓN, S. PSMAGE: Balanced map generation for StarCraft. **2013 IEEE Conference on Computational Intelligence in Games (CIG)**, p. 1–8, 2013.

WIKLUND, M. et al. Evaluating educational games using facial expression recognition software: measurement of gaming emotion. **Proceedings of the European Conference on Games Based Learning, Steinkjer**, p. 605–612, 2015.

WOLF, M. **An Intelligent Artificial Player for the Game of Risk**. Thesis (PhD) — Darmstadt University of Technology, 2005.

APÊNDICE A — INSTRUÇÕES PARA UM JOGADOR HUMANO

Os métodos básicos a serem implementados para qualquer jogador, incluindo jogador humano e artificial, são listados abaixo:

- *playAllocation(gameState)*: Responsável por retornar o território escolhido para alocar uma tropa na fase de alocação.
- *playAddUnits(gameState)*: Responsável por retornar o território escolhido para adicionar uma tropa em fases de ataque.
- *addUnitsInContinent(gameState, continent)*: Responsável por retornar o território escolhido para adicionar uma tropa a um continente ao possuir bônus de continente.
- *addUnitsInTerritory(gameState, territory)*: Responsável por retornar o território para adicionar uma tropa ao possuir bônus de território.
- *playExchangeCards(gameState)*: Responsável por retornar as três cartas escolhidas para a troca de cartas por tropas.
- *playAttack(gameState)*: Responsável por retornar os dois territórios escolhidos para a fase de ataque, sendo um o território de ataque e outro o território de defesa.
- *playMoveUnits(gameState)*: Responsável por retornar os dois territórios escolhidos para a fase de fortificação, sendo um o território de origem e outro o território de destino.

Seguindo o fluxo de jogo de uma partida, o primeiro movimento a ser feito é a alocação de tropas. A sequência de comandos dados pelo computador e pelo jogador seguem na Figura A.1. São alocados 3 territórios para cada jogador neste exemplo. Além de exibir o mapa, são exibidos também os territórios disponíveis em linha de comando. O jogador digita o território escolhido, neste exemplo em sequência: 10, 11 e 12.

Após isso, a fase de ataque começa com a adição de tropas nos territórios, conforme a Figura A.2. Os territórios disponíveis para a adição de tropas são mostrados e o jogador adiciona uma tropa por vez. No exemplo, são adicionadas 3 tropas nos territórios 15, 16 e 9. Se houver, os bônus de continente são adicionados como mostrado na Figura A.3. O comando mostra a quantidade de tropas disponíveis, o continente em que podem ser adicionadas e os territórios disponíveis. No exemplo, são adicionadas tropas nos territórios 10, 11 e 9

Quando é possível trocar cartas, o processo é feito como mostrado na Figura A.4. No exemplo, são trocadas as cartas 1, 2 e 3 e, com isso, são ganhas 2 tropas para adicionar

Figura A.1 – Sequência de comandos na fase de alocação.

```

Iniciando jogo
Fase de alocação de tropas
Território escolhido pelo jogador Player1: 5
Escolha um território para colocar um exército, territórios disponíveis:
0 1 2 3 4 6 7 8 9 10 11 12 13 14 15 16
Território escolhido: 10
Território escolhido pelo jogador Human: 10
Território escolhido pelo jogador Player1: 7
Escolha um território para colocar um exército, territórios disponíveis:
0 1 2 3 4 6 8 9 11 12 13 14 15 16
Território escolhido: 11
Território escolhido pelo jogador Human: 11
Território escolhido pelo jogador Player1: 8
Escolha um território para colocar um exército, territórios disponíveis:
0 1 2 3 4 6 9 12 13 14 15 16
Território escolhido: 12
Território escolhido pelo jogador Human: 12

```

Fonte: A Autora

Figura A.2 – Sequência de comandos na fase de ataque, adicionando tropas.

```

Você tem 3 tropas para distribuir. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13 14 15 16
Território escolhido: 10
Você tem 2 tropas para distribuir. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13 14 15 16
Território escolhido: 10
Você tem 1 tropas para distribuir. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13 14 15 16
Território escolhido: 9

```

Fonte: A Autora

Figura A.3 – Sequência de comandos na fase de ataque, adicionando bônus de continente.

```

Você tem 3 tropas para distribuir no continente 2. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13
Território escolhido: 10
Você tem 2 tropas para distribuir no continente 2. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13
Território escolhido: 11
Você tem 1 tropas para distribuir no continente 2. Escolha um território para colocar um exército, territórios disponíveis:
9 10 11 12 13
Território escolhido: 9

```

Fonte: A Autora

em qualquer território.

As batalhas da fase de ataque ocorrem como mostrado na Figura A.5. O jogador responsável pelo ataque utiliza o máximo de tropas possível em seu ataque. O comando mostra os territórios disponíveis para realizar o ataque juntamente com os territórios disponíveis para serem atacados. No exemplo, o território 14 ataca o 8 e, no próximo comando, é possível ver que o ataque foi bem-sucedido. Estas mudanças são acompanhadas também das mudanças na exibição do mapa. Se o jogador não deseja realizar ataque,

Figura A.4 – Sequência de comandos na troca de cartas.

```
Você tem 3 cartas para trocar. Escolha 3 cartas para trocar, cartas disponíveis:
Carta 0: Território 5 artilharia
Carta 1: Território 7 cavalaria
Carta 2: Território 13 cavalaria
Se não deseja trocar cartas, digite 0 0 0
Cartas escolhidas <n1 n2 n3>: 0 1 2

Você recebeu 2 tropas para distribuir
```

Fonte: A Autora

pode digitar a sequência <0 0> e a fase de ataque acaba.

Figura A.5 – Sequência de comandos na fase de ataque, atacando o oponente.

```
Escolha um território para atacar e um território para ser atacado, territórios disponíveis:
9 -> 1
10 ->
11 ->
12 ->
13 ->
14 -> 8
15 ->
16 ->

Caso não queira atacar, digite 0 0
Territórios escolhidos <ataque defesa>: 14 0
Escolha um território para atacar e um território para ser atacado, territórios disponíveis:
8 -> 5 7
9 -> 1
10 ->
11 ->
12 ->
13 ->
14 ->
15 ->
16 ->

Caso não queira atacar, digite 0 0
Territórios escolhidos <ataque defesa>: 0 0
```

Fonte: A Autora

Por fim, a fase de fortificação acontece como mostrado na Figura A.6. O comando mostra os territórios de origem disponíveis e, para cada um, os territórios destinos disponíveis. No exemplo, a escolha foi por mover uma tropa do território 14 ao 8 e encerrar a fase, utilizando a sequência <0 0>.

Figura A.6 – Sequência de comandos na fase de fortificação.

```
Escolha um território para mover os exércitos e um território para receber os exércitos, territórios disponíveis:
8 -> 14
9 -> 10 11 12
10 -> 9 12
11 -> 9 12 13
12 -> 9 10 11 13
13 -> 11 12 14
14 -> 8 13 15 16
15 -> 14 16
16 -> 14 15

Caso não queira mover exércitos, digite 0 0
Territórios escolhidos <origem destino>: 14 8
Escolha um território para mover os exércitos e um território para receber os exércitos, territórios disponíveis:
8 -> 14
9 -> 10 11 12
10 -> 9 12
11 -> 9 12 13
12 -> 9 10 11 13
13 -> 11 12 14
14 -> 8 13 15 16
15 -> 14 16
16 -> 14 15

Caso não queira mover exércitos, digite 0 0
Territórios escolhidos <origem destino>: 8 8
```

Fonte: A Autora