

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
COORDENADORIA DO CURSO DE ENGENHARIA DE SOFTWARE

LINCOLN MAGALHÃES COSTA

**UMA ABORDAGEM BASEADA EM BUSCA PARA  
GERAÇÃO DE TIMES EM JOGOS MOBA**

TRABALHO DE CONCLUSÃO DE CURSO

DOIS VIZINHOS

2019

LINCOLN MAGALHÃES COSTA

**UMA ABORDAGEM BASEADA EM BUSCA PARA  
GERAÇÃO DE TIMES EM JOGOS MOBA**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Software, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Francisco Carlos Monteiro Souza

DOIS VIZINHOS

2019



## TERMO DE APROVAÇÃO

**Uma abordagem baseada em busca para geração de times em jogos MOBA**

por

**Lincoln Magalhaes Costa**

Este Trabalho de Conclusão de Curso foi apresentado em 25 de Novembro de 2019 como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software. O(a) candidato(a) foi arguido(a) pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Francisco Carlos Monteiro Souza  
Presidente da Banca

---

Rafael Gomes Mantovani  
Membro Titular

---

Rodolfo Adamshuk Silva  
Membro Titular

\* A Folha de Aprovação assinada encontra-se na Coordenação do Curso

## AGRADECIMENTOS

Primeiramente gostaria de agradecer aos meus pais, pelo amor, incentivo e apoio incondicional. Mesmo estando fisicamente distantes, se fizeram presentes em todas as etapas dessa caminhada e a tornaram mais leve e tranquila.

Agradeço a minha namorada Bárbara, por todo o apoio, amor e compreensão. Você sempre esteve comigo nos melhores e nos piores momentos. Muito obrigado meu amor!

Aos meus amigos por todo o suporte durante a pesquisa e por compartilhar os desafios e vivências da graduação, em especial Alex, Prunzel, Casani, Lohmann, Molinete, Behne, Ricardo e Elio, que foram meus irmãos durante o curso. Vida longa à Faculdade Carinhosa.

Ao Prof. Dr. Francisco Carlos Souza, pela orientação, apoio e confiança durante os meses em que trabalhamos juntos. Foi um prazer inenarrável poder aprender com uma pessoa como você.

Por fim, agradeço a todos os funcionários da UTFPR e a todos os professores do meu curso pela alta qualidade do ensino oferecido.

## RESUMO

Costa, L. M. UMA ABORDAGEM BASEADA EM BUSCA PARA GERAÇÃO DE TIMES EM JOGOS MOBA. 74 f. Trabalho de Conclusão de Curso – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2019.

*Multiplayer Online Battle Arena* e *League of Legends* são, respectivamente, um dos gêneros e jogos eletrônicos mais jogados atualmente. Este é um estilo de jogo no qual os competidores são separados em dois times com o objetivo de destruir a base inimiga. Um dos processos mais importantes de LoL é a seleção dos campeões que serão utilizados durante a partida, já que isso envolve a estratégia da equipe e diversos fatores e variáveis que são importantes para o desenrolar da partida, como os atributos e habilidades de cada personagem. Por possuir múltiplas variáveis, é considerado um problema complexo e que possibilita a aplicação de técnicas de busca e de inteligência artificial para encontrar melhores resultados e evitar erros. Neste contexto, o principal objetivo deste projeto é prover uma abordagem capaz de facilitar as decisões nos processos que antecedem o início de uma partida de *League Of Legends*. Para isso, foi realizado um estudo à respeito dos conceitos relacionados ao projeto por meio de uma busca e revisão de literatura, e técnicas para análise de dados foram investigadas. Além disso, foi obtida uma base de dados com informações dos campeões e de seus atributos. Ela foi utilizada pela abordagem para gerar equipes e avaliá-los de acordo com as composições obtidas. Os resultados mostram que essa foi uma boa abordagem para o contexto, já que a média de *fitness* das três estratégias analisadas foi de 88.38%, com a melhor delas alcançando 96.87% em uma das configurações utilizadas. O tempo de execução também foi satisfatório, visto que esses resultados foram alcançados em execuções que duraram menos de 1 segundo.

**Palavras-chave:** *League Of Legends*, Composição de Times, Algoritmos de Busca

## ABSTRACT

Costa, L. M. A SEARCH-BASED APPROACH FOR TEAM GENERATIONS IN MOBA GAMES. 74 f. Trabalho de Conclusão de Curso – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2019.

In recent years, Multiplayer Online Battle Arena and League of Legends are the most played genre and game. This is a genre which competitors are separated into two teams facing each other in a common objective, that generally involves destroying the opponent base. One of the most important steps in LoL is the selection of the champions who are will be used in the match, since this involves many factors and variables that are important to the match such as attributes and abilities of each character. Due to several variables, it is considered a complex problem and can be handled with search-based and artificial intelligence techniques, because they are known to achieve good results in complex applications. In this context, the main objective of this project is to contribute with the players and professionals, providing an approach to facilitate the decisions made during the picks and bans before a League of Legends game. Thus, we conducted a study about the concepts regarding the project through a search and review of the literature. In addition, a database with champions information and their attributes were obtained it's going to be used by the algorithms to generate teams and evaluate them according to their compositions. Results shows that this was a good approach for this context, since the fitness average of the three analyzed strategies was 96.87% using one of the configurations. The execution time was also satisfactory, since those results was reached in executions that last less than one second.

**Keywords:** League Of Legends, Team Composition, Searching Algorithms

## LISTA DE FIGURAS

FIGURA 1	–	Mapa de League of Legends	16
FIGURA 2	–	Composição de uma equipe em League of Legends	16
FIGURA 3	–	Representação da Subida da Encosta	19
FIGURA 4	–	Fluxograma do algoritmo PSO	22
FIGURA 5	–	Fluxo do algoritmo genético	24
FIGURA 6	–	Exemplificação do <i>crossover</i>	25
FIGURA 7	–	Representação da mutação	26
FIGURA 8	–	Quantidade de estudos por jogo	29
FIGURA 9	–	Evidência dos estudos por país	30
FIGURA 10	–	Quantidade de estudos por ano	30
FIGURA 11	–	Representação do Mobator	35
FIGURA 12	–	Representação dos campeões nas respectivas rotas	36
FIGURA 13	–	Exemplar dataset	38
FIGURA 14	–	Cruzamento	40
FIGURA 15	–	Mutação	41
FIGURA 16	–	Compatibilidade da composição de <i>Poke</i> com o meta da versão 9.20	48
FIGURA 17	–	Compatibilidade da composição de <i>Hard Engage</i> com o meta da versão 9.20	48
FIGURA 18	–	Compatibilidade da composição de Team Fight com o meta da versão 9.20	49
FIGURA 19	–	Change de utilizar o Mobator	49
FIGURA 20	–	Acompanhamento do cenário competitivo	50
FIGURA 21	–	Compatibilidade da composição de <i>Poke</i> com o meta do patch 9.20	51
FIGURA 22	–	Compatibilidade da composição de <i>Hard Engage</i> com o meta do patch 9.20	52
FIGURA 23	–	Probabilidade de utilização da composição	53
FIGURA 24	–	Compatibilidade da composição de Team Fight com o meta do patch 9.20	53
FIGURA 25	–	Valor de fitness alcançado utilizando a $cfg_1$ para cada estratégia.	54
FIGURA 26	–	Valor de fitness alcançado utilizando a $cfg_2$ para cada estratégia.	55
FIGURA 27	–	Valor de fitness alcançado utilizando a $cfg_4$ para cada estratégia.	56
FIGURA 28	–	Composição gerada utilizando a estratégia de <i>Hard Engage</i> .	56
FIGURA 29	–	Composição gerada utilizando a estratégia de <i>Poke</i> .	56
FIGURA 30	–	Composição gerada utilizando a estratégia de <i>Team Fight</i> .	57
FIGURA 31	–	Média de tempo de acordo com $cfg_4$ para cada estratégia.	57

## LISTA DE TABELAS

TABELA 1	–	Associação dos termos PSO com analogia biológica .....	21
TABELA 2	–	Fontes de busca eletrônica e seus endereços .....	28
TABELA 3	–	Visão geral dos estudos primários .....	28
TABELA 4	–	Atributos iniciais utilizados pela função de <i>fitness</i> .....	42
TABELA 5	–	Atributos finais utilizados pela função de <i>fitness</i> .....	43
TABELA 6	–	Design do experimento .....	46
TABELA 7	–	Resultados do teste de Kruskal Wallis .....	55
TABELA 8	–	Resultados do teste de Kruskal Wallis para o contexto de tempo ....	58
TABELA 9	–	Comparação das estratégias com <i>cfg<sub>4</sub></i> .....	58



## LISTA DE SIGLAS

AG	Algoritmo Genético
AoS	Aeon of Strife
CPL	Cyberathlete Professional League
DOTA	Defense of the Ancients
ESL	Electronic Sports League
HC	Hill Climbing
IA	Inteligência Artificial
KeSPA	Korea e-Sports Association
kNN	k-nearest neighbor
LoL	League of Legends
MOBA	Multiplayer Online Battle Arena
PSO	Particle Swarm Optimization
QPs	Questões de Pesquisa
RL	Revisão da Literatura
RPG	Role Playing Game

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1	OBJETIVOS	11
1.2	ESTRUTURA DA MONOGRAFIA	12
<b>2</b>	<b>ASPECTOS CONCEITUAIS</b>	<b>13</b>
2.1	JOGOS ELETRÔNICOS E GÊNERO MOBA	14
2.2	ALGORITMOS BASEADOS EM BUSCA	19
2.2.1	Otimização por Enxame de Partículas	21
2.2.2	Algoritmo Genético	22
<b>3</b>	<b>REVISÃO NARRATIVA DA LITERATURA</b>	<b>27</b>
3.1	OBJETIVO DA PESQUISA	27
3.2	SÍNTESE E ANÁLISE DOS RESULTADOS	28
3.3	AMEAÇAS À VALIDADE	33
<b>4</b>	<b>MOBATOR - UMA ABORDAGEM AUTOMÁTICA PARA COMPOSIÇÃO DE TIMES EM LOL</b>	<b>34</b>
4.1	PROPOSTA DE PESQUISA	34
4.2	COLETA DE DADOS	36
4.3	MODELAGEM	38
4.4	FUNÇÃO DE AVALIAÇÃO	39
<b>5</b>	<b>AVALIAÇÃO EXPERIMENTAL</b>	<b>44</b>
5.1	PLANEJAMENTO	44
5.2	SURVEY	46
5.2.1	Survey versão I	47
5.2.2	Estratégia <i>Poke/Pusher</i>	47
5.2.3	Estratégia <i>Hard Engage</i>	48
5.2.4	Estratégia Team Fight	48
5.2.5	Survey versão II	50
5.2.6	Estratégia <i>Poke/Pusher</i>	51
5.2.7	Estratégia <i>Hard Engage</i>	51
5.2.8	Estratégia Team Fight	51
5.3	RESULTADOS	52
<b>6</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>60</b>
6.1	CONTRIBUIÇÕES	60
6.2	LIMITAÇÕES	61
6.3	TRABALHOS FUTUROS	61
	<b>REFERÊNCIAS</b>	<b>62</b>
	Apêndice A – SURVEY VERSÃO I	65
	Apêndice B – SURVEY VERSÃO II	70

## 1 INTRODUÇÃO

Seguindo a constante evolução da indústria de jogos eletrônicos, o crescimento do *Multiplayer Online Battle Arena* (MOBA) (MORA-CANTALLOPS; SICILIA, 2018) no mundo dos games é bastante notável. Atualmente, existem diversos campeonatos da modalidade como DOTA 2 e League of Legends com premiações milionárias.

A maioria dos MOBAs possuem as mesmas mecânicas e objetivos, onde ao entrar na partida, os jogadores escolhem um personagem dentre os disponíveis. Cada personagem (ou herói) possui habilidades, vantagens e desvantagens específicas. É importante que os personagens sejam escolhidos de forma estratégica, de acordo com o papel que cada jogador desempenha na partida.

Em seguida, os times (compostos geralmente por 5 integrantes) são colocados em uma arena virtual na qual o objetivo é destruir a base adversária. Para isso, os jogadores precisam avançar estrategicamente no mapa, balanceando ataque e defesa. Assim, ao longo da partida, os jogadores evoluem seus personagens conforme matam unidades inimigas ou neutras (que não pertencem a nenhum time, geralmente encontradas em locais fora das rotas principais do mapa), compram itens e destroem estruturas adversárias. As partidas terminam quando um dos times avança o suficiente e consegue destruir a principal estrutura adversária, situada na base.

Um ponto importante que pode ser analisado de forma automática e antes mesmo do início da partida é o *draft*: processo de seleção dos personagens que serão utilizados durante o confronto. Atualmente, a elaboração e planejamento dessa etapa é feita de forma manual pelos técnicos e analistas das equipes, utilizando cadernos e anotações referentes à equipe e aos adversários.

Esse processo é de extrema importância, pois cada personagem possui características próprias e também atributos que se sobressaem ou que são ofuscados (os chamados *counterpicks*) quando confrontados por outros personagens. Esses fatores são somados às atualizações frequentes que acontecem no jogo, onde campeões são fortificados e/ou

enfraquecidos para manter o balanceamento da jogabilidade. O processo de escolher campeões fortificados e aproveitar a sinergia entre eles é chamado de meta.

Além de ser uma atividade de extrema importância para o sucesso do time, a seleção dos personagens para o jogo é uma atividade complexa e difícil de automatizar. Por essa razão, alguns trabalhos têm abordado com frequência o uso de algoritmos de aprendizado de máquina e técnicas de estatísticas para tomadas de decisão como Oliveira et al. (2017) e Tanuar et al. (2018).

Embora essa tarefa seja considerada complexa, ela pode ser tratada como um problema de busca, isto é, é possível a utilização de técnicas de buscas e algoritmos da Inteligência Artificial (IA) para solucionar esse problema. Técnicas de busca em IA consistem em um grupo de algoritmos que podem ser aplicados em problemas de grande complexidade, os quais são dificilmente solucionados por técnicas convencionais, como por exemplo problemas de otimização. Nesse sentido, o uso da análise de dados somado às técnicas de buscas podem auxiliar na tomada de decisão e na geração de times mais competitivos, baseando-se nos históricos de partidas dos principais jogadores do mundo e nas taxas de vitória de cada campeão.

A proposta desse trabalho desenvolve um gerador de times inteligente para jogos MOBA, provendo uma abordagem, denominada de MOBA AID (acrônimo para *Artificial Intelligence Draft*), para auxiliar à tomada de decisão para os profissionais envolvidos no processo de *draft*, com um foco principal em League of Legends. Considerando os personagens existentes no jogo, os *counterpicks* (campeões que se destacam contra outros) e os níveis de habilidade de cada jogador, tem-se mais de 25 milhões de possibilidades para um único time.

## 1.1 OBJETIVOS

O objetivo geral desse trabalho é propor uma abordagem inteligente capaz de auxiliar os jogadores e profissionais envolvidos no processo de geração de composições de equipes em League of Legends. O propósito da solução é permitir que os problemas complexos apresentados anteriormente neste capítulo sejam resolvidos, utilizando conceitos e tecnologias atuais em relação a técnicas de busca e metaheurística.

Baseando-se no objetivo geral foram definidos os seguintes objetivos específicos que foram desenvolvidos:

1. Estudos e levantamentos dos aspectos conceituais;
2. Estudo da linguagem de programação javascript;
3. Investigação de técnicas para análise de dados e estatísticas do jogo;
4. Criação da base de dados;
5. Criação de uma função de avaliação baseada em informações do jogo;
6. Desenvolvimento de um protótipo funcional; e
7. Avaliação experimental e estudos comparativos;

## 1.2 ESTRUTURA DA MONOGRAFIA

A monografia está dividida em 4 principais capítulos: os aspectos conceituais são apresentados no Capítulo 2, onde se tem uma base dos principais conceitos utilizados durante o desenvolvimento desta pesquisa. Já no Capítulo 3 é apresentada a revisão de literatura, onde são exibidos estudos relacionados ao contexto deste trabalho. No Capítulo 4 é apresentada a proposta deste projeto e os resultados obtidos são apresentados no Capítulo 5. Por fim, no Capítulo 6, são expostas as considerações finais.

## 2 ASPECTOS CONCEITUAIS

Os jogos eletrônicos têm se tornado uma das principais fontes de receita da indústria de entretenimento mundial. Segundo Ubisoft (2018), por exemplo, o mercado dos *games* já movimenta mais dinheiro que a indústria da música, do cinema e que o consumo de livros.

Recentemente, diversos pesquisadores têm focado em desenvolverem abordagens inteligentes para criar estratégias de jogos (LOOI et al., 2018) (CONLEY; PERRY, 2013) (OLIVEIRA et al., 2017). Dentre as abordagens propostas, técnicas de ciência de dados e da inteligência artificial tem sido bastante utilizadas, as quais serão abordadas de forma mais completa nas próximas seções.

Ciência de dados (do inglês, *data science*) é um campo interdisciplinar de investigação de dados que utiliza métodos científicos e técnicas avançadas de análise de dados e inteligência artificial para resolver problemas reais. Uma característica interessante da área de ciência de dados é a conexão criada entre o mundo acadêmico e profissional. Na academia, muito se aprende à respeito de métodos científicos, estruturação de pesquisas e análises, na indústria, por sua vez, existe um forte apelo pela rápida resolução de problemas práticos nos diversos setores das organizações.

No contexto de jogos, diversas abordagens têm utilizado análise de dados, porém, atualmente, poucas utilizam técnicas de busca e pouquíssimas ou nenhuma usam a combinação dessas duas técnicas. A união delas pode prover uma abordagem inteligente que agregue informações e estatísticas reais com uma forte técnica de resolução de problemas.

Nesse sentido, este capítulo visa a apresentar os conceitos fundamentais referentes ao trabalho de graduação a ser realizado. Na Seção 2.1, são sintetizados os principais conceitos e terminologias referentes ao estudo e as técnicas e algoritmos baseados em busca são apresentados na Seção 2.2.

## 2.1 JOGOS ELETRÔNICOS E GÊNERO MOBA

A história dos games começou em 1947 (JR; MANN, 1948). Considerado o primeiro jogo da história, ele era um simulador de mísseis inspirado nos radares da segunda guerra mundial. Pode-se separar os jogos eletrônicos em três plataformas principais: videogames, computadores e aparelhos móveis. Segundo Rogers (2012), existem diversos gêneros de jogos usados para descrever um estilo de jogabilidade, destacando-se os jogos de ação, esportes, tiro e estratégia, onde estão os jogos MOBA como League of Legends e DOTA 2.

A mistura entre a grande expansão do mercado e dos consumidores de jogos obrigou as empresas a fomentarem ainda mais esse nicho. O resultado, que tem se mostrado muito bem acertado, foi a criação de competições e transmissões que ocorrem no mundo todo, nos mais diferentes níveis de profissionalização. A estruturação da categoria ficou conhecida como *eSports* (EGGERT MARC HERRLICH; MALAKA, 2015), abreviação para *electronic Sports* ou esportes eletrônicos.

O gênero MOBA surgiu dentro de um jogo de estratégia chamado StarCraft. Ele foi lançado em 1998 pela Blizzard e trouxe um recurso inédito chamado StarEdit, modo que permitia a customização de mapas e diversos elementos do jogo. A primeira criação a trazer confrontos entre equipes que precisavam destruir a estrutura principal do time adversário foi *Aeon of Strife (AoS)*. A popularidade de AoS cresceu com o tempo, mas o lançamento de *Warcraft 3: Reign of Chaos* dividiu a atenção dos jogadores de estratégia e trouxe diversas novidades na customização de StarCraft, já que o jogo também era da Blizzard. Com as atenções divididas, outras customizações surgiram e dentre elas estava o cenário original de *Defence Of The Ancients (DOTA)*, que ganhou grande popularidade após a geração de uma versão mais estável.

Entre 2004 e 2005, a organização dos jogadores para a criação de times e campeonatos ocorria de forma online, por meio de fóruns. Aos poucos, equipes e ligas profissionais interessaram-se no crescimento da comunidade de DOTA e em 2005 a *Cyberathlete Professional League (CPL)* fez sua primeira competição do MOBA no território asiático, enquanto a *Electronic Sports League (ESL)* organizava os campeonatos online pela Europa e pela América do Norte.

Os passos mais marcantes do nascimento do esporte eletrônico foram registrados na Coreia do Sul, com a criação da *Korea e-Sports Association (KeSPA)*, um órgão criado em 2000 pelo ministério coreano da cultura, esporte e turismo com o objetivo de aproximar a categoria com os esportes tradicionais. No contexto atual, frequentemente encontramos

transmissões de campeonatos online em canais esportivos da TV fechada no Brasil como Esporte Interativo e SporTV, despertando ainda mais interesse nos *games*.

Para este estudo, o foco será o League of Legends, um MOBA (*Multiplayer Online Battle Arena* ou Arena de Batalha de Multijogadores) desenvolvido e publicado pela empresa estadunidense Riot Games desde 2009, ano em que foi lançado. Segundo o 2018 Year In Review <sup>1</sup>, publicado pela SuperData, a receita do League of Legends no ano foi de cerca de U\$1,4 bilhões de dólares, colocando o jogo na terceira colocação dos mais rentáveis de 2018. Além disso, o canal da Riot Games possui o seu canal na lista dos mais assistidos da Twitch.

MOBA é um gênero de jogos eletrônicos que mistura elementos de estratégia, ação e *Role Playing Game (RPG)*. Antes de cada partida, o jogador seleciona qual personagem irá representá-lo dentro da arena, o qual também é chamado de campeão em LoL. Após o início do jogo, as duas equipes devem disputar por recursos, controle de território e objetivos secundários para alcançar a vitória, assegurada após a destruição da base do time inimigo. Cada personagem evolui conforme acumula recursos e elimina os adversários, aprendendo novas habilidades, fortalecendo as mesmas e comprando itens.

A maioria dos MOBAs possuem as mesmas mecânicas e objetivos. Ao entrar na partida, os jogadores escolhem um personagem dentre os disponíveis, esse processo é conhecido como *draft*. Cada personagem (herói ou campeão) possui habilidades, vantagens e desvantagens específicas. É importante que os personagens sejam escolhidos de forma estratégica, de acordo com o papel que cada jogador desempenha na partida.

Em seguida, os times (compostos geralmente por 5 integrantes) são colocados em uma arena virtual (exposta na Figura 1) na qual o objetivo é destruir a base adversária. Para isso, os jogadores precisam avançar estrategicamente no mapa, balanceando ataque e defesa. Assim, ao longo da partida, os jogadores evoluem seus personagens conforme matam unidades inimigas ou neutras (que não pertencem a nenhum time, geralmente encontradas em locais fora das rotas principais do mapa), compram itens e destroem estruturas adversárias. As partidas terminam quando um dos times avança o suficiente e consegue destruir a principal estrutura adversária, situada na base (FERRARI, 2013).

---

<sup>1</sup>Relatório exclusivo e de acesso limitado contendo diversas informações à respeito do mercado de jogos eletrônicos.

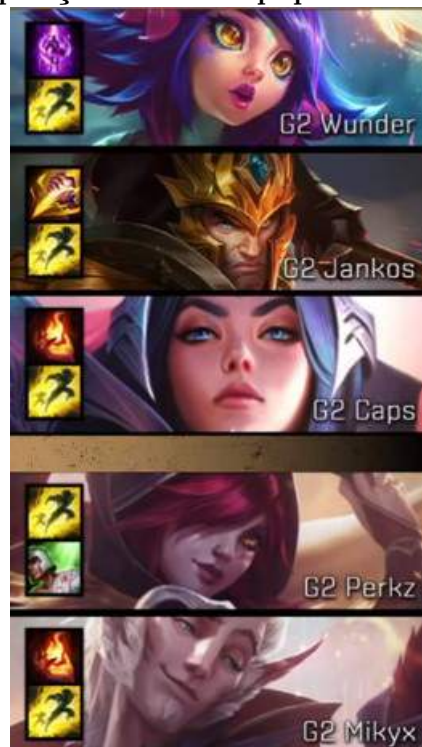


Figura 1: Mapa de League of Legends



Fonte: Adaptada de Riot Games <sup>2</sup>

Figura 2: Composição de uma equipe em League of Legends



Fonte: Adaptada de Riot Games <sup>3</sup>

Compreendendo-se as características básicas de League of Legends, busca-se entender as suas informações e etapas complexas, as quais são pouco exploradas e a academia tem buscado aprofundar os seus estudos para melhor compreendê-las e aperfeiçoá-las, como a relação entre a composição de uma equipe (ilustrada na Figura 2) e

a sua performance (WOLFE et al., 2005), (SALAS; COOKE; ROSEN, 2008). O processo de estruturação de uma composição para a partida é um dos exemplos complexos que existem em jogos MOBAs.

Até o momento em que este trabalho é escrito, existem 146 campeões disponíveis para escolha dos jogadores. Em League, esses campeões são distribuídos em 5 diferentes papéis onde podem atuar:

1. **Carregador:** também conhecido como *Attack Damage Carry* (ADC), seu principal papel é "carregar" a equipe, conquistando objetivos (como torres e campeões neutros) e abates de forma rápida e eficiente. Esses campeões possuem alto dano e mobilidade porém são extremamente frágeis, fazendo com que o jogador precise demonstrar um bom posicionamento durante a partida;
2. **Suporte:** principal companheiro do ADC, o suporte tem como função proteger o carregador e o time como um todo, auxiliando no controle de visão do mapa e fazendo com que os demais campeões consigam obter os recursos que precisam;
3. **Caçador:** assim como o Suporte, o caçador possui o objetivo de auxiliar a equipe a conquistar recursos e objetivos. Por manter-se na selva, um espaço neutro entre os dois times, os caçadores conseguem obter esteroides e amplificadores e também são capazes de organizar emboscadas para campeões inimigos;
4. **Meio:** atuando ao lado do ADC, os campeões da rota do meio também possuem um peso importante como fonte de dano da equipe. Geralmente, os campeões dessa rota são magos e/ou assassinos, e costumam ter como objetivo abater o carregador da equipe adversária; e
5. **Topo:** esses campeões atuam sozinhos na rota superior. Entre suas responsabilidades estão: o controle dos objetivos desse lado do mapa (Barão e Arauto do Vale) e presença na linha de frente durante as lutas em equipe. Por estar em uma rota isolada, é imprescindível que o jogador consiga obter vantagem em ouro e abates para manter o adversário acoado;

Apesar das definições, cada um dos papéis possui adaptações de acordo com a estratégia adotada pela equipe, sendo possível trocar papéis dentre as rotas e/ou abdicar de um deles para cumprir um estilo de jogo. Além disso, como citado no Capítulo 1, o meta também é um fator de influência que pode fazer com que papéis não sejam seguidos de

forma tradicional, já que campeões fortificados têm a sua prioridade de escolha aumentada e podem ser selecionados em rotas não tradicionais.

A cooperação é um fator essencial em League of Legends. Existem campeões com grande sinergia entre si e combinar esses personagens faz com que as suas características e habilidades sejam fortificadas. Em larga escala, utilizar composições de equipe onde existe sinergia entre todos os campeões faz com que o time possua uma força singular para vencer a partida. Existem diversas composições em League mas algumas aparecem com maior frequência:

- Composições de ***Team Fight*** possuem campeões com dano elevado e com habilidades que são aplicáveis em área, possibilitando causar dano a vários inimigos ao mesmo tempo.
- A estratégia de ***Poke*** ou ***Push*** conta com campeões de grande alcance de ataque e habilidades, já que o seu objetivo é levar objetivos e garantir abates de forma segura, evitando lutas em equipe e mantendo o time inimigo a uma distância segura.
- As composições de ***Hard Engage*** focam em forçar lutas em equipe. Geralmente contam com campeões com controles de grupo (habilidades que enfraquecem o inimigo) e com alto dano, já que o seu principal objetivo é criar lutas decisivas e abater 1 ou 2 adversários.

Dito isso, observa-se que a importância desse procedimento é altamente elevada, pois cada personagem possui características próprias e atributos que se sobressaem ou que são ofuscados quando confrontados por outros personagens<sup>4</sup>. Soma-se a esses fatores, a habilidade pessoal de cada jogador, já que existem atletas profissionais que são mundialmente conhecidos por possuírem extrema habilidade com heróis específicos.

Por tratar-se de um problema complexo e de grande importância em seu meio, estudos recentes como Oliveira et al. (2017), Conley e Perry (2013) e Tanuar et al. (2018) têm abordado com frequência o uso de aprendizagem de máquina e ciência de dados para tomadas de decisão e composição de equipes em jogos MOBA. Entretanto, nenhuma abordagem focou na geração de equipes considerando diferentes restrições como *counterpick*, habilidade do jogador, sinergia entre os campeões e técnicas inteligentes. Dadas as suas características, a ciência de dados também seria útil e somaria para o contexto, tendo

---

<sup>4</sup>Essas vantagens são chamadas de *counterpicking*, que é o ato de selecionar um personagem que possui vantagem sobre os heróis que já foram escolhidos pelo adversário.

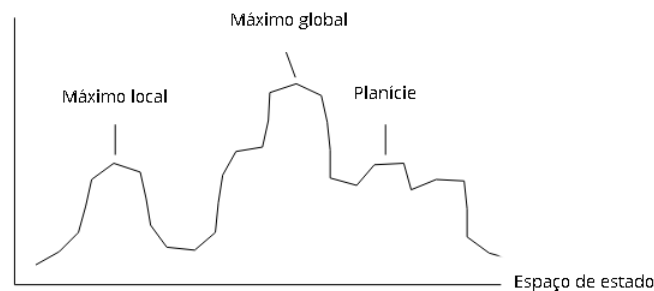
em vista que é um campo interdisciplinar de investigação de dados que utiliza métodos científicos e técnicas avançadas de análise de dados e inteligência artificial para resolver problemas reais.

## 2.2 ALGORITMOS BASEADOS EM BUSCA

Em ciência da computação, técnicas ou algoritmos de busca são técnicas de inteligência artificial aplicadas a problemas de otimização e alta complexidade que não são resolvidos com técnicas convencionais de programação (RUSSELL; NORVIG, 2009). A complexidade de um problema está diretamente ligada ao tamanho do seu espaço de busca, ou seja, a quantidade de possibilidades disponíveis para serem analisadas. Em termos gerais, é um algoritmo que recebe um problema como entrada e tem a solução do problema como saída.

Uma das técnicas de busca utilizadas é a Subida da Encosta (do inglês, *Hill Climbing - HC*). De acordo com Russell e Norvig (2009) seu nome deriva da semelhança da abstração de uma escalada no Monte Everest em meio de um nevoeiro denso durante uma crise de amnésia.

**Figura 3: Representação da Subida da Encosta**  
Função objetivo



**Fonte: Adaptada de Russell e Norvig (2009)**

Na Figura 3 é apresentada uma representação de um espaço de busca unidimensional, onde a elevação corresponde a função objetivo e o algoritmo de busca tem como objetivo buscar o máximo global. O máximo local e a planície também são apresentados na imagem, onde o primeiro trata-se de um pico mais alto em comparação com os seus vizinhos mas ainda menor do que o máximo global e o segundo representa uma área plana onde não há uma evolução do algoritmo, pois todos os vizinhos possuem uma avaliação similar.

O objetivo é chegar no pico mais alto apenas examinando os locais mais próximos (vizinhos) e esquecendo os caminhos anteriores, e terminar quando alcançar um pico em

que nenhum vizinho tenha o valor mais alto.

Comumente, a técnica de Subida da Encosta encontra boas soluções, porém Rich e Knight (2001) descrevem três tipos de problemas:

1. Falsos pontos de máximo, chamados de máximos locais (como apresentado na Figura 3) que forçam o algoritmo a um grande retrocesso para alcançar o máximo verdadeiro;
2. Condições onde não há uma diferença relevante para a escolha, ou seja, para qualquer avanço não há registro de melhora nos índices classificatórios; e
3. Os chamados "cumes", região interessante do espaço de busca que não podem ser alcançadas em um único movimento por estarem a uma distância considerável do ponto de partida;

Existem maneiras de contornar os problemas citados acima. Soluções que devem ser levadas em consideração:

1. **Reinício aleatório:** o algoritmo executa uma sequência de buscas a partir de estados iniciais gerados de forma aleatória. Isso evitaria o mesmo de sempre encontrar planaltos e máximos locais; e
2. **Limitadores:** executar a busca até que um certo limitador seja atingido. Esse limitador pode estar vinculado ao número de iterações ou até que não haja uma melhora significativa nos resultados encontrados;

No Algoritmo 1, observa-se o laço de repetição apresentado por Russell e Norvig (2009) e a verificação que o algoritmo faz para checar os pontos vizinhos a cada iteração.

---

**Algorithm 1** Subida da Encosta

---

**procedure** HC(problema) **return** uma solução

**Entradas:** problema

**variáveis locais:** solução corrente, solução vizinha

*corrente*  $\leftarrow$  *criarNo*(*estadoInicial*[problema])

**while** critério de parada não for atingido **do**

*vizinho*  $\leftarrow$  um sucessor da solução corrente com uma melhor avaliação

**if** (*valor*[*vizinho*]  $\leq$  *valor*[*corrente*]) **then return** *estado*[*corrente*]

**end if**

**end while**

**end procedure**

---

### 2.2.1 OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS

Um dos algoritmos que tem se destacado para fugir dos problemas expostos no HC é o algoritmo de otimização por enxame de partículas (do inglês, *Particle Swarm Optimization - PSO*), desenvolvido por Kennedy e Eberhart (1995), que implementa uma representação da interação entre indivíduos (partículas) e o grupo em que está inserido (enxame).

O PSO foi criado após a observação do comportamento social de cardumes e bandos de aves, onde as partículas (consideradas como soluções candidatas) comportam-se como animais que se beneficiam de algum tipo de comportamento social por não possuírem uma grande inteligência individual. Isso faz com que aumentem as suas chances de acasalamento, facilita a busca por comida, reduz a probabilidade de ataque por predadores, entre outros benefícios. Essa relação entre o contexto técnico e a analogia biológica está explicada na Tabela 1.

**Tabela 1: Associação dos termos PSO com analogia biológica**

<b>Termos PSO</b>	<b>Significado baseado em analogia biológica</b>
Enxame	Bando de pássaros
Espaço de busca	Área sobrevoada em busca de alimento
Posição	Localização de cada pássaro
Solução Ótima	Localização do pássaro que encontrou alimento
<i>Fitness</i>	Função de avaliação
Pbest	Melhor posição conhecida pelo pássaro
Gbest	Melhor posição conhecida pelo grupo

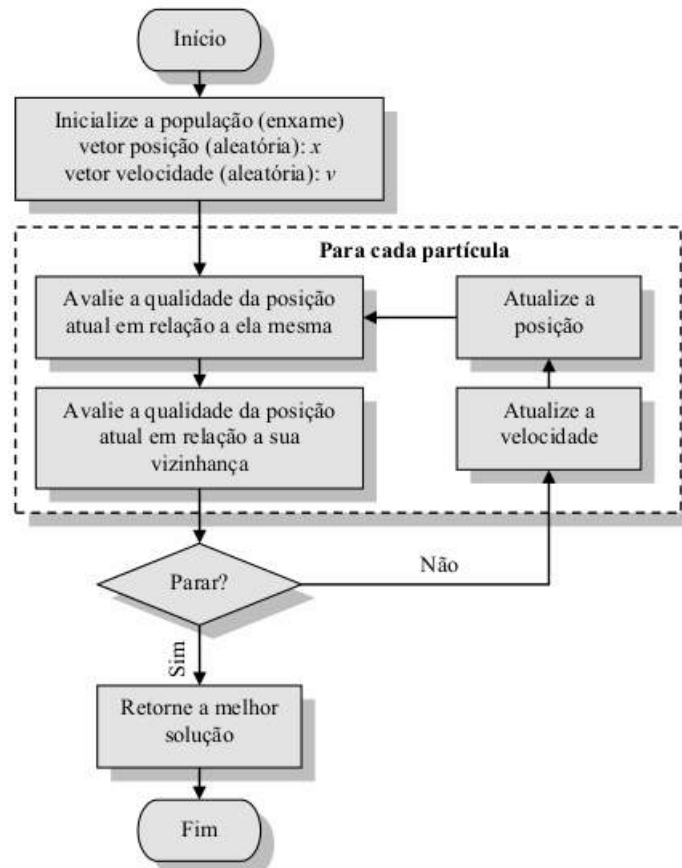
**Fonte: Adaptada de Souza (2017a)**

Um importante diferencial dessa técnica é o fato de que tais partículas percorrem um espaço n-dimensional e são capazes de armazenar informações passadas das coordenadas no espaço de busca com o valor da melhor solução encontrado (*pbest*) e compartilhar com os demais indivíduos (*gbest*).

A movimentação dos indivíduos dentro do espaço de busca é norteadas por três fatores principais: 1) Posição; 2) Velocidade; 3) Melhor posição encontrada pela partícula; 4) Melhor posição encontrada pelo grupo. Na Figura 4, é apresentado o fluxograma básico do funcionamento do PSO e no algoritmo 2 são exibidas as instruções para a execução do algoritmo.

Observa-se que após o começo da execução, a população é inicializada com vetores de posição e velocidade aleatórios. Uma repetição é aplicada para todas as partículas,

Figura 4: Fluxograma do algoritmo PSO



Fonte: Nascimento (2010)

fazendo com que elas avaliem a qualidade das suas posições atuais e da posição atual em relação a sua vizinhança. Caso a vizinhança tenha uma qualidade inferior, ela não irá parar. Ao invés disso, sua velocidade e posição serão atualizadas e essa interação se repetirá até que a vizinhança tenha uma qualidade superior em comparação com a posição atual; quando isso ocorrer, a melhor solução será retornada.

### 2.2.2 ALGORITMO GENÉTICO

Uma outra técnica de computação evolucionária que opera sobre uma população de soluções candidatas em paralelo é o Algoritmo Genético (AG), proposta por Holland (1992). Esse tipo de algoritmo é muito eficiente para buscar soluções ótimas em uma grande variedade de problemas, pois não impõem muitas das limitações encontradas nos métodos tradicionais.

Os Algoritmos Genéticos (AG) são métodos de otimização e busca inspirados no princípio da seleção natural e sobrevivência dos mais aptos, declarado pelo filósofo e

---

**Algorithm 2** Otimização por enxame de partículas
 

---

```

procedure PSO(partículas, posições, velocidades)
  for cada partícula do
    Iniciar partícula;
  end for
  while critério de parada não for atingido do
    for cada partícula do
      Calcular o valor de fitness  $f(p)$ ;
      Atualizar o melhor valor de fitness da partícula até o momento;
      if ( $f(p)$  melhor que  $p_{best}$ ) then
        Definir o valor atual como o novo  $p_{best}$ ;
      end if
    end for
    Atualizar o melhor valor de fitness da população até o momento;
    Definir o  $g_{best}$  com o melhor valor de fitness de todas as partículas;
    for cada partícula do
      calcular velocidade e posição;
    end for
  end while
end procedure

```

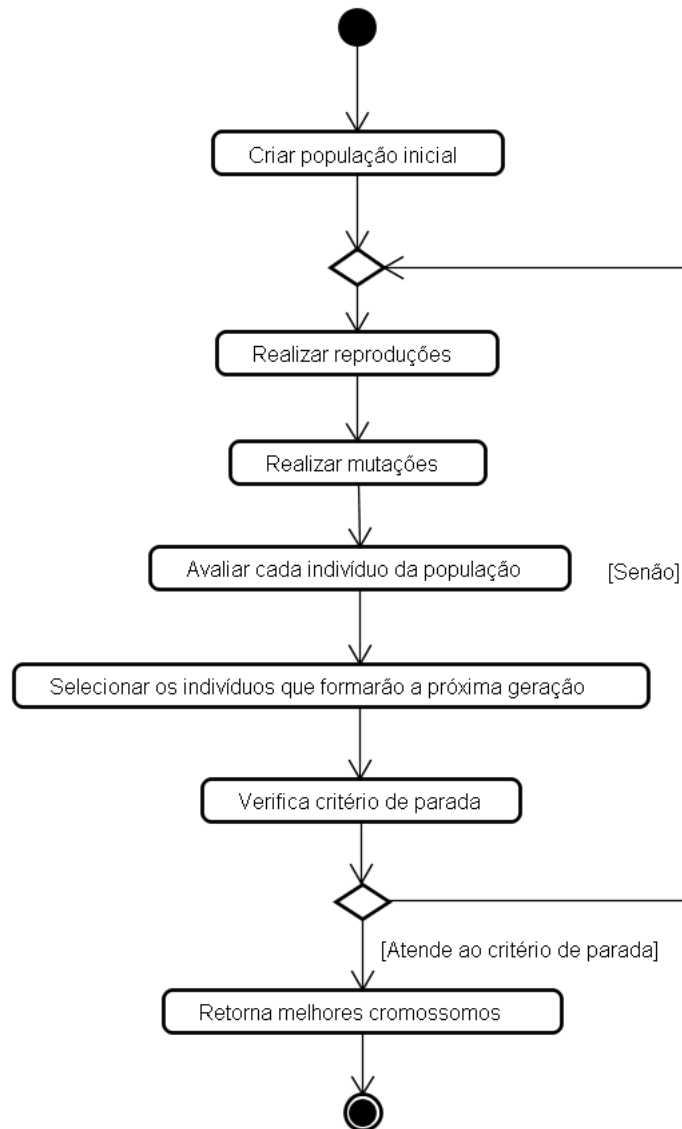
---

naturalista Charles Darwin em seu livro A Origem das Espécies. De acordo com Darwin (2018), “Quanto melhor um indivíduo se adaptar ao seu meio ambiente, maior será sua chance de sobreviver e gerar descendentes”.

Na Figura 5, tem-se, de forma simplificada, os passos que são executados pelo algoritmo genético. A representação contém uma população inicial e uma função de avaliação (também chamada de *fitness*) como entrada para uma função e tem, como saída, o melhor indivíduo encontrado após a aplicação do processo evolutivo.



Figura 5: Fluxo do algoritmo genético



Fonte: Luque (2010)

Para melhor compreensão a respeito do seu funcionamento, faz-se necessária a definição de alguns termos:

1. **População:** Conjunto de soluções candidatas para resolver um problema; a população inicial geralmente é aleatória.
2. **Indivíduo:** Representação da solução do problema que se pretende solucionar.
3. **Geração:** Representa um ciclo ou uma iteração do algoritmo genético.
4. **Função de avaliação de (*fitness*):** Determina a qualidade de um indivíduo como

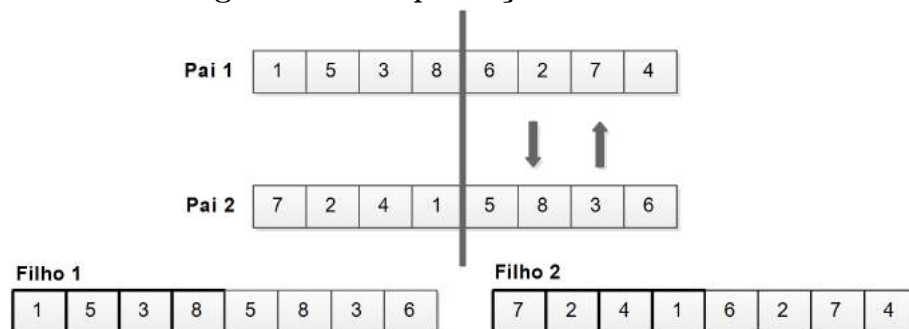
solução do problema. Deve-se atentar para sua determinação, pois será a responsável por mapear informações para solucionar o problema.

5. **Critério de parada:** Processo que tem como objetivo avaliar quando o processamento do algoritmo deve ser finalizado. Pode-se levar em conta o número de gerações criadas, o alcance da função objetivo com um valor ótimo ou a convergência na mesma (quando a mesma não apresenta uma melhoria significativa).
6. **Seleção:** O método de seleção age simulando o mecanismo de seleção natural existente em espécies biológicas.
7. **Reprodução:** Etapa onde os indivíduos mais adaptados, ou seja, com melhores valores segundo a função *fitness*, tem a chance de perpetuar o seu material genético através dos operadores genéticos como cruzamento (também chamado de *crossover*).

O princípio básico dos operadores genéticos é transformar a população através de sucessivas gerações, estendendo a busca até obter um resultado satisfatório. Os operadores genéticos são necessários para que a população diversifique-se e mantenha características de adaptação adquiridas pelas principais gerações anteriores.

O *crossover* funciona de maneira similar a mutação, a sua diferença está no fato de que diversos cromossomos são selecionados de forma aleatória ao invés de um único item, como se tem na Figura 14. Esse material genético é combinado com o de outro indivíduo, gerando dois outros indivíduos. O seu principal objetivo é extrair os melhores genes de cada indivíduo e combiná-los de maneira a possuir as melhores características. A ideia do operador de cruzamento é tirar vantagem do material genético presente na população.

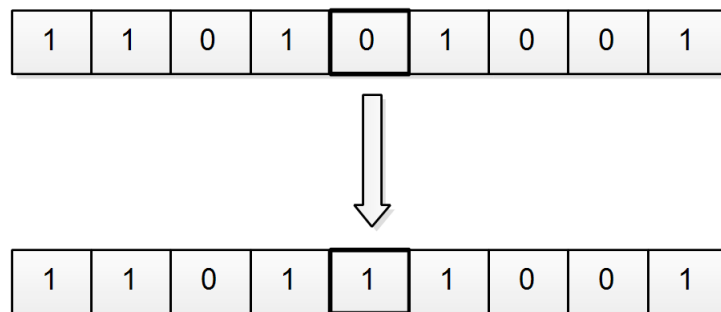
**Figura 6: Exemplificação do *crossover***



Fonte: Souza (2017a)

Na mutação, os valores contidos nos genes de um cromossomo são trocados de forma aleatória por outro valor válido. Esse processo possibilita que sejam incluídas informações que não estiveram presentes em nenhum dos indivíduos e também garante a diversidade das características dos indivíduos da população. Além disso, a mutação também viabiliza uma busca aleatória no AG, permitindo que mais pontos do espaço de busca sejam avaliados. Esse operador está ilustrado na Figura 7.

**Figura 7: Representação da mutação**



Fonte: Souza (2017a)

No Algoritmo 3, tem-se uma outra representação do fluxo do AG, agora focando em sua codificação. Observa-se o funcionamento explicado nesta seção, onde o AG inicia e avalia a população. Enquanto o critério de parada não for atingido, os melhores indivíduos serão selecionados para a próxima população. Serão aplicados cruzamentos e mutações e os novos indivíduos gerados serão avaliados. Após o atingimento do critério de parada, o fluxo é finalizado.

---

### Algorithm 3 Algoritmo Genético

---

```

procedure AG
  Iniciar a população;
  while critério de parada não for atingido do
    Aplicar cruzamento e mutação;
    Avaliar a população;
    Selecionar indivíduos para próxima população;
  end while
  Retornar os melhores cromossomos;
end procedure

```

---

### 3 REVISÃO NARRATIVA DA LITERATURA

Esse capítulo refere-se a Revisão da Literatura (RL) conduzida para a elaboração deste trabalho. Para isso, foi adotada como estratégia uma revisão narrativa a fim de apresentar e discutir o estado da arte do tema abordado (PRODANOV; FREITAS, 2013). Essa estratégia foi escolhida pois, mesmo pesquisando em bases diferentes das citadas neste estudo, poucos resultados foram encontrados em buscas iniciais. Dessa forma, a partir de Questões de Pesquisa definidas, foram analisados trabalhos na literatura para interpretação e análise crítica e, ao final, a comparação do presente projeto com algumas das referências identificadas.

Este capítulo está organizado da seguinte forma: a estratégia juntamente com o objetivo de pesquisa são apresentados na Seção 3.1. Na Seção 3.2, é apresentada a síntese e a análise dos resultados obtidos.

#### 3.1 OBJETIVO DA PESQUISA

A RL iniciou-se com a definição das Questões de Pesquisa (QPs), essenciais para identificar evidências disponíveis na literatura referentes a dois contextos principais: *i*) técnicas de geração e avaliação de equipes; e *ii*) predição de resultados.

**QP1: Quais técnicas/abordagens têm sido utilizadas para a geração de equipes em jogos MOBA?**

**QP2: Quais abordagens ou técnicas têm sido utilizadas para a predição de resultados em jogos MOBA?**

Após a definição das QPs, uma *string* de busca foi formulada. Ao aplicá-la em bases como *IEEE Xplore* e *ACM Digital Library*, poucos estudos foram retornados. Visando encontrar um maior número de resultados, as questões de pesquisa foram aplicadas nas fontes de busca eletrônica descritas na Tabela 2 e também em repositórios de conferências da área de estudo e em bases como *arXiv* e Google Acadêmico.

**Tabela 2: Fontes de busca eletrônica e seus endereços**

Fontes de busca	Endereço eletrônico
IEEE Xplore	<i>www.ieeexplore.ieee.org</i>
arXiv	<i>www.arxiv.org</i>
Google Acadêmico	<i>scholar.google.com</i>
ACM	<i>dl.acm.org</i>

**Fonte: Autoria própria**

Como resultado da busca realizada, 10 estudos foram selecionados após serem lidos na íntegra e avaliados conforme a sua similaridade com o tema deste trabalho.

### 3.2 SÍNTESE E ANÁLISE DOS RESULTADOS

Com os estudos retornados pela busca e sua aderência ao tema analisado, realizou-se a atividade de síntese após a extração das informações relevantes de cada um deles. Primeiramente, serão apresentadas informações gerais a respeito dos estudos e depois as análises e resultados de cada Questão de Pesquisa serão exibidos.

Na Tabela 3 é apresentada a distribuição dos 10 estudos selecionados. Na primeira coluna, tem-se a identificação do estudo e o seu ano de publicação está na coluna Referência. Na coluna Base de Busca são apresentadas as bases de onde os estudos foram extraídos.

**Tabela 3: Visão geral dos estudos primários**

Estudo	Referência	Base de Busca
E1	(CONLEY; PERRY, 2013)	Google Acadêmico
E2	(SUZNJEVIC; MATIJASEVIC; KONFIC, 2015)	IEEE
E3	(KIM et al., 2016)	ACM
E4	(WANG, 2016)	arXiv
E5	(OLIVEIRA et al., 2017)	Google Acadêmico
E6	(SOUZA, 2017b)	Google Acadêmico
E7	(SAPIENZA; GOYAL; FERRARA, 2018)	arXiv
E8	(CHENG et al., 2019)	IEEE
E9	(LOOI et al., 2018)	arXiv
E10	(TANUAR et al., 2018)	IEEE

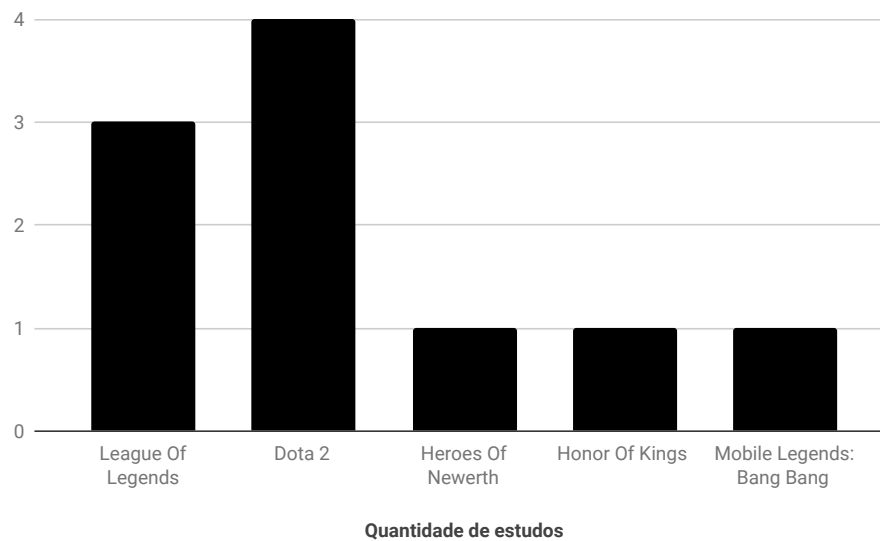
**Fonte: Autoria própria**

É importante destacar que 70% dos estudos apresentados foram publicados nos últimos 5 anos, mas todos foram escritos há menos de uma década, o que assegura a atualidade dos temas pesquisados.

Nesta revisão, foram identificados, também, os jogos que mais têm sido base para

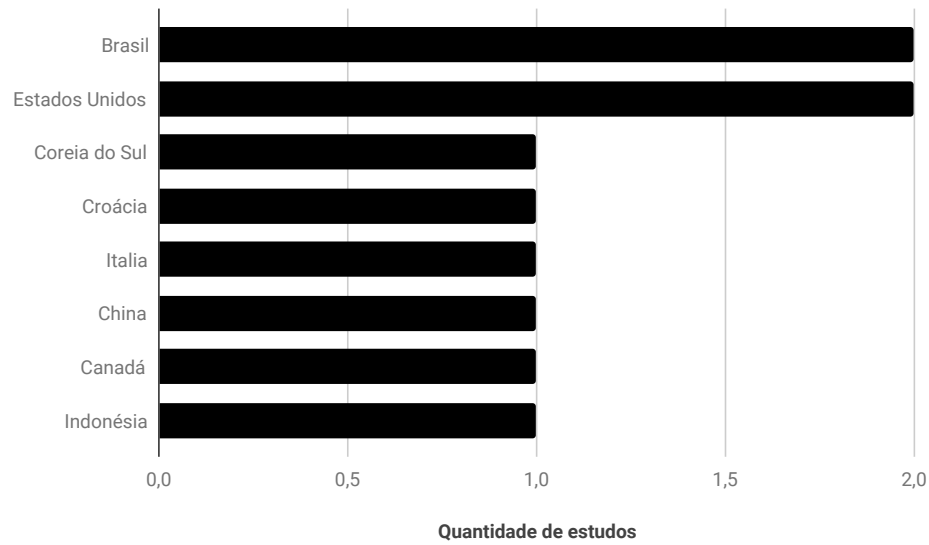
essas pesquisas. A classificação dos jogos é baseada no número de estudos produzidos com relação ao jogo. Conforme a Figura 8, Dota 2 e League of Legends destacaram-se, tendo 4 e 3 estudos, respectivamente, dentre os selecionados. Pode-se criar uma conexão entre a relevância desses dois títulos e os números mostrados na Seção 2.1 dos Aspectos Conceituais, onde nota-se a proeminência desses dois jogos.

**Figura 8: Quantidade de estudos por jogo**



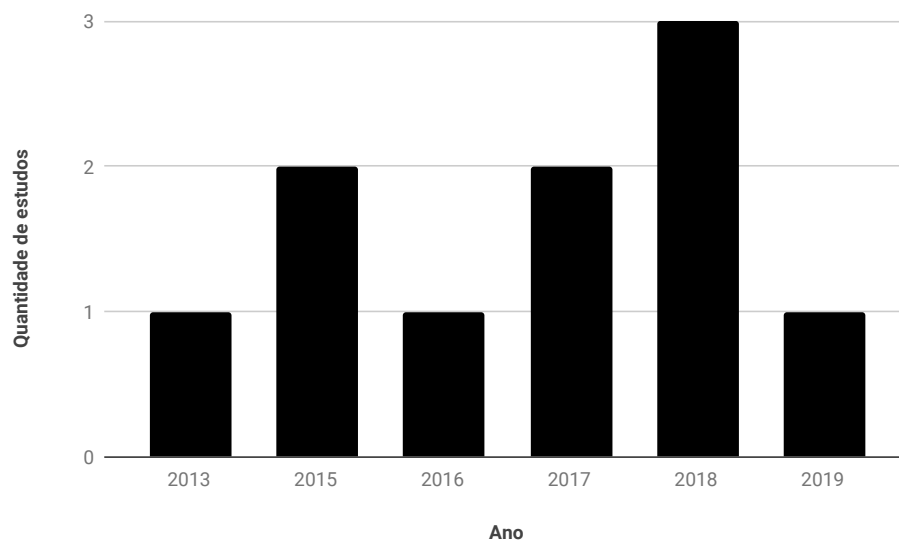
**Fonte: Autoria própria**

Os pesquisadores dos Estados Unidos e do Brasil são os que possuem maior número de publicações nessa área, com 2 estudos cada. Coreia do Sul, Croácia, Itália, Indonésia, China e Canadá possuem somente um estudo publicado na área, como se vê na Figura 9.

**Figura 9: Evidência dos estudos por país**

**Fonte: Autoria própria**

Uma informação interessante que se extrai dos estudos selecionados é referente ao seu ano de publicação. Conforme vê-se na Figura 10, a maioria dos estudos foi escrito nos últimos 3 anos, somam-se 6 publicações nos anos de 2017, 2018 e 2019, afirmando que o tema é atual e de interesse dos pesquisadores.

**Figura 10: Quantidade de estudos por ano**

**Fonte: Autoria própria**

Somente Conley e Perry (2013) abordou algoritmos de busca em sua pesquisa,

onde a busca gulosa foi utilizada na recomendação de heróis. Os autores não deram muitas informações a respeito da utilização da técnica de busca, porém sua abordagem tinha quatro etapas principais:

1. É criado um vetor de características da partida;
2. São criados novos exemplares de vetores de característica, cada um deles com diferentes heróis candidatos adicionados ao vetor original;
3. Execução do algoritmo para computar a probabilidade de vitória de cada vetor de características descrito no passo 2, calculando a probabilidade do time ser vencedor;
4. Ordena os candidatos de acordo com a probabilidade de vitória das recomendações avaliadas.

Além disso, também foram utilizadas técnicas de predição, regressão logística e algoritmo *k-nearest neighbors* (*kNN*) ou algoritmo do vizinho mais próximo. O objetivo do seu estudo era gerar um algoritmo de recomendação de campeões em *Defense of the Ancient 2* (*Dota 2*), um MOBA similar a League of Legends. Seus resultados foram satisfatórios, conseguindo 67% de acurácia durante a validação do algoritmo. Esse foi um dos estudos que serviram como resposta para a primeira Questão de Pesquisa, onde buscava-se encontrar quais abordagens e técnicas têm sido utilizadas para a geração de equipes em jogos MOBA.

Um outro modelo que se destacou dentre os estudos selecionados, foi as redes neurais, utilizadas por Wang (2016) e Sapienza, Goyal e Ferrara (2018). O primeiro estudo tem como objetivo funcionar como um preditor de vitórias com base no processo de seleção e banimento de campeões em DOTA 2. Durante a validação do algoritmo, uma acurácia de 61% foi alcançada. Já Sapienza, Goyal e Ferrara (2018), procuram analisar a composição das equipes e otimizá-las por meio da relação entre cooperação (no sentido literal), times, recomendação de jogadores para a formação de equipes e performance individual de cada um dos jogadores. Seus resultados mostraram a viabilidade de um novo formato de recomendação de companheiros de equipe, sugerindo que a transferência de habilidades e melhoria de performance pode ser prevista com redes neurais. Ambos os estudos somaram no objetivo de responder a QP2.

Cheng et al. (2019) abordou o problema de forma diferente e utilizou validação cruzada com o uso de Xgboost<sup>1</sup> comparando jogos dos mesmos jogadores em diferentes

---

<sup>1</sup>O XGBoost é uma eficiente implementação de código aberto do algoritmo baseado em árvores com



espaços de tempo para estudar a eficiência de composição de equipe no MOBA Honor Of Kings. Em seu estudo, analisou as diferentes funções e papéis (*roles*, em inglês) dentro do jogo e a influência de cada uma delas para medir a performance de uma equipe. Durante o estudo, 3 principais variáveis foram analisadas pelos autores:

- Linha de base (do inglês, *baseline*): segundo estudos citados pelo autor, o humor do jogador varia de acordo com os horários e o dia da partida, fazendo com que a performance seja influenciada;
- Informações do jogador: classificação do jogador de acordo com a função que será desempenhada;
- Propriedades das funções dos personagens: possíveis combinações de papéis dentro do jogo;

Essas variáveis foram exploradas pelos autores do estudo em 4 cenários diferentes:

- Predição de vitória da equipe;
- Predição de vitória da equipe onde ambos os times tem rankings (ou classificações) similares;
- Predição de rendição; e
- Predição de abuso;

Os resultados obtidos mostraram que, para o experimento em questão e em todos os cenários, o fator mais relevante é a junção de todas essas variáveis (denominada, pelos autores, como *total*). As propriedades das funções dos personagens ficou no segundo local de maior relevância em todos os cenários.

Performance de equipes e a influência da sua formação para alcançar a vitória também foi a base de estudo de Kim et al. (2016), onde métodos qualitativos e quantitativos foram utilizados para analisar tais fatores em *League of Legends*. Para as variáveis qualitativas, 10 jogadores coreanos de diferentes classificações e ranqueamentos no MOBA foram entrevistados para o estudo. Durante as respostas percebeu-se que os jogadores de níveis mais baixos atribuem vitórias a sorte e ao destaque de um único jogador, e não ao trabalho em equipe.

---

aumento de gradiente. O aumento de gradiente é um algoritmo de aprendizagem supervisionada que tenta prever com precisão uma variável de destino.

Em contrapartida, os jogadores mais habilidosos relataram que a vitória está diretamente ligada a personagens competentes de acordo com a composição que está sendo montada pela sua equipe. As implicações das informações obtidas pelos autores mostram que jogadores experientes possuem maior facilidade para lidar com os modelos propostos pelo *game*, enquanto os novatos terão que desenvolver isso gradativamente, podendo tornar a sua experiência de jogo frustrante.

Após a condução dessa revisão, pode-se analisar o quão interessante é o problema em questão, devido a sua complexidade, sua relevância para o mercado dos jogos eletrônicos e por tratar-se de um assunto atual. Com os estudos encontrados, também foi possível explorar as soluções propostas pelos pesquisadores ao redor do mundo, estudando os resultados obtidos nos estudos de cada um deles.

Apesar dos resultados positivos, numerosos e atualizados obtidos nas bases de buscas eletrônica, a literatura carece de conteúdo relacionado ao tema proposto neste trabalho. Durante a revisão literária, percebeu-se que pouco tem sido realizado nesse contexto.

### 3.3 AMEAÇAS À VALIDADE

São ameaças identificadas no decorrer do desenvolvimento da pesquisa. O processo de seleção de estudos relacionados foi aplicado nas principais bases da área porém pode não contar com todos os estudos que seriam relevantes para este estudo. Além disso, como já citado, em alguns casos, os autores não especificaram por completo determinadas informações relevantes (como a construção de suas funções de *fitness*), dificultando a compreensão dos seus estudos. Dentre as ameaças, também pode-se citar as constantes atualizações em League of Legends, já que é necessário que a base de dados mantenha-se sempre atualizada de forma que os resultados encontrados pelo algoritmo sejam sempre os mais pertinentes.

## 4 MOBATOR - UMA ABORDAGEM AUTOMÁTICA PARA COMPOSIÇÃO DE TIMES EM LOL

Este capítulo tem como objetivo principal descrever a proposta deste trabalho de conclusão de curso, estabelecendo os principais objetivos do projeto, a metodologia e os fundamentos e as tecnologias que serão utilizadas para seu desenvolvimento.

Conforme descrito no Capítulo 1, neste projeto será empregado o conceito de técnicas de busca (mais especificamente algoritmo genético) para otimizar processos envolvendo a geração de equipes em League of Legends e melhorar os resultados obtidos por essas equipes e pelos profissionais envolvidos. Dessa forma, é possível utilizar técnicas de otimização baseadas em busca direcionada por uma determinada função de avaliação projetada para o problema abordado.

A abordagem proposta é apresentada neste capítulo, iniciando-se na Seção 4.1, onde são apresentadas suas principais características. Na Seção 4.2, são apresentados os dados utilizados e informações sobre os mesmos, a modelagem é descrita na Seção 4.3 e a função de avaliação é explicada na Seção 4.4.

### 4.1 PROPOSTA DE PESQUISA

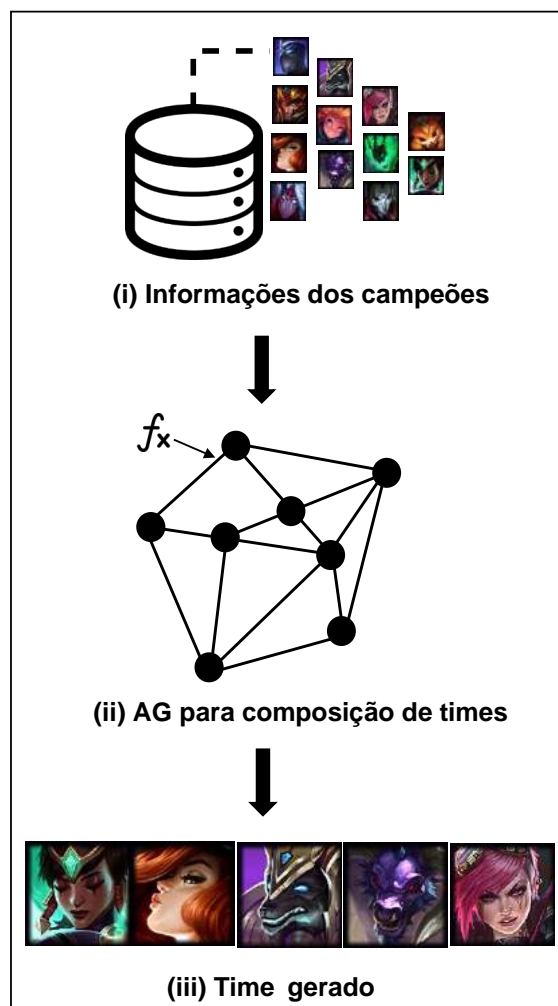
Quando se pensa em processamento elevado de números, logo imagina-se um grande computador realizando todos os cálculos necessários para chegar no resultado esperado. Apesar disso, essa não é a realidade envolvida nas possibilidades que tange o processo de geração de composições de equipes, onde os técnicos fazem as análises utilizando papéis e canetas e sem grande auxílio da tecnologia, possibilitando o aparecimento de erros.

O League of Legends possui, no momento em que esse trabalho é escrito (versão 9.20), 146 campeões disponíveis para serem selecionados por uma equipe de 5 jogadores. Considerando esse número para a formação de uma equipe, tem-se um número extremamente elevado de possibilidades, aumentando assim o espaço de busca que a abordagem deve

percorrer para encontrar uma solução otimizada.

Considerando esse cenário para a geração de equipes, é possível observar que se trata de um problema complexo e de difícil resolução, tornando imprescindível o uso de técnicas mais robustas para alcançar boas soluções e evitar possíveis erros que aconteceriam caso a tarefa fosse realizada de maneira manual. O presente trabalho apresenta a utilização de técnicas de busca para encontrar a melhor composição possível para diferentes estratégias de jogo, baseando-se em estatísticas reais.

**Figura 11: Representação do Mobator**



Fonte: Adaptado de Costa, Souza e Souza (2019)

Conforme apresentado na Figura 11, o Mobator emprega atributos de campeões e do meta em uma função matemática para avaliar os times gerados. A representação de dados utilizada na abordagem é um arquivo JSON com informações de todos os 146 campeões e esta é apresentada com maior clareza na Seção 4.2. Atributos como *roles*,

papéis no jogo e taxa de vitória são utilizados para avaliar a adequação da equipe gerada e conduzir o processo de busca para um ótimo resultado no espaço de busca. Adicionalmente, uma validação foi criada para garantir que um campeão não irá se repetir dentro de uma composição e outra para garantir que os campeões respeitam os papéis básicos de League, conforme exibido na Figura 12, onde os campeões da representação do Mobator estão espalhados em suas respectivas rotas.

**Figura 12: Representação dos campeões nas respectivas rotas**



Fonte: Adaptada de Riot Games <sup>1</sup>

Diversos jogos online utilizam cooperação, um mecanismo presente em sistemas do mundo real, em diferentes escalas para melhorar a experiência de jogo, dando aos jogadores uma oportunidade de jogar e cooperar com pessoas de qualquer lugar do planeta. Essa relação fica clara em jogos como League, onde os *players* precisam selecionar o seu campeão e jogar pensando nos companheiros de equipe e na composição proposta, já que todos terão responsabilidades específicas no decorrer da partida.

## 4.2 COLETA DE DADOS

Baseando-se nas evidências encontradas na literatura, optou-se pela utilização de um *dataset* para a execução do Algoritmo Genético. Os dados dos campeões de League of Legends foram obtidos em uma base em formato JSON disponibilizada em um repositório online pela Riot Games, produtora do jogo.

Além dessas informações, também foram necessários ajustes manuais no *dataset* para garantir que informações relacionadas a características empíricas e estatísticas (como papéis que o personagem desempenha durante a partida e taxa de vitória, respectivamente) fossem consideradas pela função de avaliação. Para obter tais informações, a ferramenta *op.gg*<sup>2</sup> foi utilizada, já que por meio dela é possível acessar páginas com detalhes sobre cada campeão e obter informações sobre os papéis que o campeão desempenha e taxa de vitória na versão 9.20 de League of Legends.

Na Figura 13 é apresentado um exemplo dos objetos presentes no conjunto de dados, tais informações também são mencionadas na Tabela 5.

---

<sup>2</sup>*Op.gg* é uma das maiores ferramentas para análise de jogos eletrônicos como League of Legends, Overwatch, Fortnite, entre outros.

Figura 13: Exemplar dataset

```
{
  "localized_name": "Thresh",
  "infos": {
    "attack": 5,
    "defense": 6,
    "magic": 6,
    "difficulty": 7,
    "winrate": {
      "support": 50.2
    }
  },
  "roles": [
    "Tank",
    "Crowd Control",
    "Hard Engage",
    "Disengage",
    "Area of Effect"
  ],
  "lanes": [
    "Support"
  ],
  "id": 118,
  "icon":
  "http://ddragon.leagueoflegends.com/cdn/9.19.1/img/champion/Thresh.png"
}
```

Fonte: Autoria própria

#### 4.3 MODELAGEM

Anos de estudo fizeram com que a técnica de Algoritmo Genético alcançasse resultados promissores nos mais diversos contextos, tais como os apresentados por Zhamri et al. (2010) e Deaven e Ho (1995). Apesar da composição de equipes não estar dentre esses, os resultados obtidos pelas abordagens nestes estudos demonstram que essa técnica

pode ser uma boa escolha para o contexto de otimização de times.

O processo de busca, no contexto de League, inicia-se com a criação randômica de equipes como população inicial e, depois disso, os times são avaliados. Após a avaliação, o AG inicia a etapa de melhoria e evolução da população por meio de operadores genéticos conforme apresentado nas Figuras 14 e 15. O funcionamento do AG para a geração de composições também é demonstrado no Algoritmo 4 e este fluxo é realizado até que o número máximo de gerações seja atingido.

---

**Algorithm 4** AG para Composições de Equipes

---

```

procedure AG
  Inicializa a população de equipes;
  Avalia os times da população;
  while não atinge o número de gerações do
    Seleciona indivíduos;
    Aplica cruzamento e mutação;
    Avalia a nova população;
  end while
end procedure

```

---

A Figura 14 apresenta um exemplo de cruzamento ocorrendo entre dois times, onde cada gene no indivíduo é um campeão diferente. Um cruzamento funciona através da troca de genes entre as soluções por meio de um ponto de corte. Após esse processo, dois novos indivíduos são criados contendo informações do Indivíduo 1 e Indivíduo 2.

O operador de mutação tem o propósito de prover diversidade à população. Ele funciona realizando uma mutação em um indivíduo, conforme exibido na Figura 15 em uma possível composição. Um gene no cromossomo é selecionado e alterado para um campeão qualquer da base de dados. Genes já existentes no cromossomo são desconsiderados.

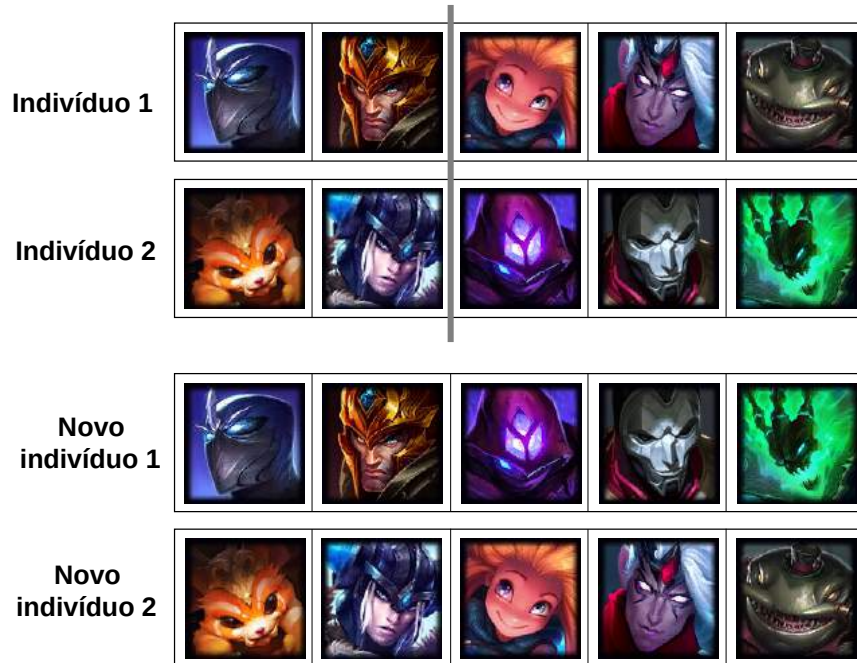
No contexto de composições de equipes em League of Legends, bons indivíduos são aqueles em que os campeões possuem sinergia entre si e que se adequam com a estratégia proposta pela equipe. Esses também foram os fatores considerados para desenvolver a função de *fitness* desta abordagem, apresentada com mais detalhes na próxima seção.

#### 4.4 FUNÇÃO DE AVALIAÇÃO

Uma função de *fitness* (ou avaliação) adequada é considerada a parte mais importante de técnicas baseadas em busca, pois ela é a responsável por guiar a exploração em busca dos melhores resultados.



Figura 14: Cruzamento



Fonte: Adaptado de Costa, Souza e Souza (2019)

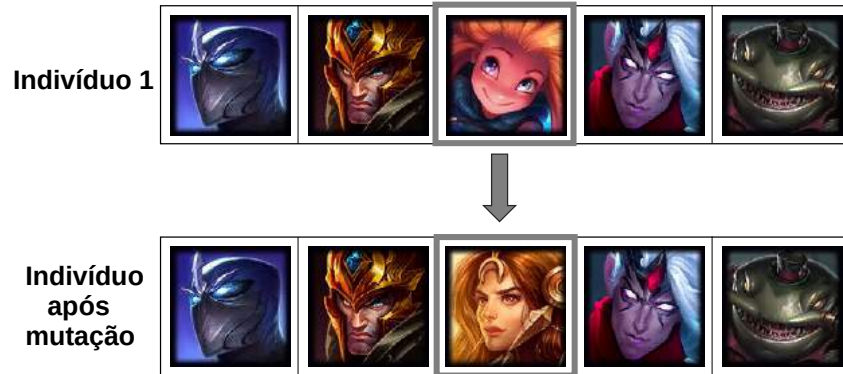
Ao jogar League of Legends, existem diversas estratégias que podem levar um time à vitória. Visando manter um ambiente balanceado, cada uma delas tem suas respectivas forças e fraquezas (assim como cada um dos campeões). Como exemplo dessa situação, podemos citar as composições de *poke* e *hard engage*. Enquanto o objetivo da primeira é causar dano à longo alcance e sem receber dano do adversário, a segunda composição foca em iniciações rápidas e certeiras, forçando lutas entre as equipes. Assim, essa abordagem foca em três estratégias de batalha: Poke (I), Hard Engage (II) e Team Fight (III), com base nisso, para esse estudo foram propostas três funções de *fitness* diferentes.

No geral, estratégias de jogo e composições de equipe são mais complexas do que simplesmente unir campeões, principalmente se considerarmos os papéis disponíveis em League. Por este motivo e para não termos um problema sem solução, foram definidas duas restrições para as composições.

A primeira delas é que todas as equipes geradas devem preencher todos os cinco papéis de jogo (carregador, suporte, caçador, meio e topo), e a segunda é que para ser considerado um time de sucesso, todos devem seguir uma das estratégias de batalha definidas no estudo (Hard Engage, Team Fight, Poke).

Formalmente, o problema de composições de time consiste na identificação de um conjunto de campeões que maximizem os seus atributos respeitando as restrições citadas

Figura 15: Mutaç o



Fonte: Adaptado de Costa, Souza e Souza (2019)

acima.

Maximiza o

$$\sum_{i=1}^n atrib_1 + atrib_2 + atrib_m \quad (1)$$

sujeito a

$$estrategia \in \{hardengage, teamfight, poke\} \quad (2)$$

$$time = (carregadorC_1, suporteC_2, topoC_3, \dots, meioC_5) \quad (3)$$

onde  $C_i$    o campe o,  $n$    o tamanho do time e   igual a 5 e  $atrib_m$  s o atributos dos campe es selecionados.

Inicialmente, foram propostas tr s fun es de avalia o considerando os atributos apresentados na primeira coluna da Tabela 4. Essas fun es iniciais consideravam os pontos de atributos de cada um dos campe es do jogo, nivelando e avaliando-os de acordo com o encaixe de cada um desses atributos nas estrat gias pretendidas.

A equa o 4 define o valor m ximo que pode ser obtido por uma estrat gia de Hard Engage por meio da fun o de avalia o, de acordo com o tamanho da equipe (TE), que   uma constante de valor 5, o maior valor de MV e DA no *dataset*. Esses atributos foram considerados inicialmente por se encaixarem no objetivo da estrat gia: causar o m ximo de dano poss vel enquanto o time possui mobilidade o suficiente para mover-se

Tabela 4: Atributos iniciais utilizados pela função de *fitness*

Atributos	Valor máximo	Estratégias
Dano de ataque (DA)	70	<i>Team Fight, Poke e Hard Engage</i>
Dano de ataque por level (DAL)	5	Team Fight
Alcance de ataque (AA)	650	Poke
Velocidade de ataque por level (VAL)	6	Poke
Pontos de vida (PV)	800	Team Fight
Velocidade de movimento (VM)	355	Hard Engage

em direção ao adversário.

$$hardEngage = (\max(DA) + \max(VM)) * TE \quad (4)$$

A equação que expressa o valor máximo de *fitness* que pode ser alcançado pela estratégia de Poke é apresentada na equação 5. Foram escolhidos tais atributos devido ao fato da estratégia buscar um alto alcance nos campeões em que a compõem.

$$poke = (\max(DA) + \max(AA) + \max(VAL)) * TE \quad (5)$$

O valor máximo que um time utilizando a estratégia de Team Fight poderia alcançar está expresso na equação 6. Esta estratégia usa estatísticas como dano de ataque, dano de ataque por level e pontos de vida para calcular o maior valor de avaliação.

$$teamFight = (\max(DA) + \max(DAL) + \max(PV)) * TE \quad (6)$$

Entretanto, após a observação dos resultados preliminares (obtidos via *survey* e avaliação empírica), constatou-se que os times resultantes não eram compatíveis com os que estavam sendo utilizados no meta, tornando-os pouco viáveis e interessantes. Assim, optou-se por considerar parâmetros que tenham maior influência no meta e que acompanhem tais mudanças, buscando composições melhores de um ponto de vista empírico.

Os atributos utilizados pela função de avaliação final são apresentados na Tabela 5.

**Tabela 5: Atributos finais utilizados pela função de *fitness***

Atributo	Descrição
Taxa de vitória	Taxa de vitória do campeão em uma respectiva <i>lane</i>
<i>Role</i>	Características sobre o campeão e suas habilidades
<i>Lane</i>	Papel que o campeão costuma desempenhar

**Fonte: Autoria própria**

A Equação 7 define o valor máximo que pode ser obtido por cada uma das três estratégias analisadas no estudo, de acordo com a média da taxa de vitória dos campeões e o fator multiplicador. Observa-se que *winrate* refere-se a taxa de vitória de um campeão em um determinado papel no jogo (as *lanes* citadas na Tabela 5), *s* faz referência a estratégia,  $C_i$  a um campeão e *n* ao tamanho de uma equipe, que constantemente será 5. Por fim, visando bonificar as equipes que mais se encaixassem no objetivo da estratégia proposta, um multiplicador foi aplicado ao fim da obtenção do valor de *fitness*. O valor inicial do multiplicador (também chamado de *roleEvidence*) é 1.0 e a cada evidência de uma *role* (ou característica) que beneficie a estratégia soma-se 0.1 ao multiplicador, podendo totalizar 1.5.

$$s = \left( \sum_{i=5}^n \text{winrate}C_i/5 \right) * \text{roleEvidence} \quad (7)$$

## 5 AVALIAÇÃO EXPERIMENTAL

No capítulo anterior foi apresentada uma proposta para a geração de times em jogos MOBA utilizando técnicas de busca, mais especificamente algoritmo genético.

Para validar a abordagem, neste capítulo é apresentada uma avaliação experimental com foco no esquema de avaliação proposto. A qualidade das composições geradas por meio da abordagem proposta também foram comparadas entre si e com composições utilizadas no jogo durante a realização da pesquisa. Além disso, duas versões de um *survey* foram conduzidas com jogadores de League of Legends de diferentes níveis para identificar a opinião dos mesmos sobre as composições geradas pela abordagem. Neste experimento foram utilizadas as funções de *fitness* com base nas estratégias de batalha e a nova formulação apresentada na Equação 7.

Na Seção 5.1, são expostos os detalhes do planejamento e os objetivos da avaliação, além das questões de pesquisa e hipóteses que devem ser validadas. Na Seção 5.1 é retratado o procedimento para condução dos experimentos e na Seção 5.2 são apresentadas as informações referentes ao *survey*. Os resultados obtidos e as discussões relacionadas a eles são apresentados na Seção 5.3.

### 5.1 PLANEJAMENTO

O modelo utilizado para estabelecer os objetivos do experimento foi o *Goal-Question-Metric* (GQM) (BASILI; WEISS, 1986), resumando estes propósitos no seguinte:

*"Analisar a abordagem proposta com o propósito de avaliar à respeito de função de fitness e tempo do ponto de vista de experimentos e questionários (survey) no contexto de composições de equipes em League of Legends."*

Assim, são consideradas as seguintes Questões de Pesquisas (QPs), divididas entre parâmetros de eficiência e eficácia:

**QP1: Para qual estratégia a abordagem para geração de composições mostra-se mais eficaz?**

Para responder a ( $QP_1$ ), a eficácia da abordagem foi avaliada utilizando os valores de *fitness* encontrados para cada uma das estratégias. O experimento foi executado 10 vezes, as médias ( $\mu F_i$ ) de cada geração da melhor configuração foram calculadas, onde  $\mu F_1$ ,  $\mu F_2$ ,  $\mu F_3$  são referente as estratégias *Poke*, *Hard Engage* e *Team Fight*. Além disso, as respostas obtidas no *survey* também serviram como argumento para responder essa Questão de Pesquisa.

Para esta QP as seguintes hipóteses foram definidas:

**H1<sub>0</sub>:** Não existe diferença entre os valores de *fitness* obtidos pelas funções  $\mu F_1$ ,  $\mu F_2$ ,  $\mu F_3$ , assim:

$$\mathbf{H1_0:} \mu F_1 = \mu F_2 = \mu F_3$$

**H1<sub>1</sub>:** Existe diferença entre os valores de *fitness* obtidos pelas funções  $\mu F_1$ ,  $\mu F_2$ ,  $\mu F_3$ , assim:

$$\mathbf{H1_1:} \mu F_1 \neq \mu F_2 \neq \mu F_3$$

**QP2: Qual estratégia é a mais eficiente para geração de composições?**

O tempo (T) é a variável considerada para responder a **QP2**. Para isso, executamos o experimento por 10 vezes, computamos os maiores valores de *fitness* obtidos e calculamos o tempo que abordagem levou para alcançar essa composição. Foram definidas as seguintes hipóteses para essa Questão de Pesquisa:

**H2<sub>0</sub>:** Não existe diferença entre o tempo de execução utilizado pela ( $\mu T_1$ ), ( $\mu T_2$ ), ( $\mu T_3$ ), assim:

$$\mathbf{H2_0:} \mu T_1 = \mu T_2 = \mu T_3$$

**H2<sub>1</sub>:** Existe diferença entre o tempo de execução utilizado pela ( $\mu T_1$ ), ( $\mu T_2$ ), ( $\mu T_3$ ), assim:

$$\mathbf{H2_1:} \mu T_1 \neq \mu T_2 \neq \mu T_3$$

### QP3: Qual é a melhor estratégia para geração de composições?

Além disso, um questionário no modelo *survey* foi desenvolvido para validar as composições geradas. Segundo Wohlin (2000), essa também é uma oportunidade para entender melhor a população de afetados pela pesquisa.

Para responder as Questões de Pesquisa, foram definidas quatro configurações diferentes para executar os experimentos. Essas configurações foram escolhidas devido aos resultados alcançados no experimento de Costa, Souza e Souza (2019). A partir desse estudo, identificou-se a necessidade de uma maior taxa de mutação para aumentar a diversidade da população e também notou-se que um baixo número de indivíduos na população proporciona uma evolução mais lenta. Para cada estratégia as três configurações foram utilizadas, as mesmas são apresentadas na Tabela 6. Nela tem-se que  $P$  é referente ao tamanho da população,  $MR$  está ligado a taxa de mutação do algoritmo e  $G$  é o número de gerações que existirão durante cada execução.

**Tabela 6: Design do experimento**

Estratégias	Configurações	Parâmetros		
		P	MR	G
Poke, Hard engage e Team fight	$cfg_1$	200	0.7	100
	$cfg_2$	200	0.7	1000
	$cfg_3$	300	0.7	100
	$cfg_4$	300	0.7	1000

## 5.2 SURVEY

Visando analisar os resultados obtidos por meio de uma ótica empírica e qualitativa, um *survey* foi desenvolvido com o objetivo de identificar a opinião de jogadores sobre os times gerados pela abordagem. O público-alvo da pesquisa é composto por jogadores de diferentes níveis (novato/casual, experiente/competitivo ou profissional), sem nenhuma amostragem específica.

Após a definição dos objetivos, definiu-se que o *survey* seria disponibilizado em formato de questionário eletrônico. O questionário foi desenvolvido utilizando a ferramenta Google Forms<sup>1</sup>. As questões foram estruturadas em seis seções e estas estão disponíveis no Apêndice A.

A primeira seção contém uma apresentação do *survey* em que é descrito o objetivo do mesmo e o seu público-alvo. A segunda seção foi desenvolvida para coletar

<sup>1</sup>É possível obter mais informações em <https://www.google.com/forms/about/>

informações que caracterizam os participantes, são elas: nome de usuário em League of Legends, elo atual<sup>2</sup>, nível de conhecimento e habilidades sobre o jogo e nível de acompanhamento do cenário competitivo de League. Esses últimos utilizaram a Escala Likert<sup>3</sup> como resposta.

As seções três, quatro e cinco são relacionadas as três estratégias apresentadas neste estudo. Nelas os participantes tiveram que analisar as composições geradas pela abordagem e avaliá-las de acordo com as perguntas realizadas. Por fim, a seção seis buscava obter sugestões e ideias para estudos futuros.

### 5.2.1 SURVEY VERSÃO I

O questionário foi distribuído em chats e fóruns do jogo. Essa distribuição ocorreu entre 24 de outubro e 08 de novembro de 2019. Após este período, os dados coletados foram analisados, visando compreender a opinião dos jogadores à respeito das composições propostas pelo Mobator.

O *survey* foi respondido por 97 pessoas, dos mais diversos níveis de jogabilidade. Dentre os dados analisados, é possível observar que a maior parte dos participantes (61.9%) se considera um jogador experiente/competitivo, 34% acredita ser novato/casual e 4.1% está em um nível profissional.

As perguntas relacionadas ao perfil básico visam apresentar informações gerais dos participantes. Dessa forma, tem-se que 41.2% dos participantes acompanham ao máximo o cenário competitivo profissional de *League* (resposta 5 em um intervalo de 1 a 5).

Segundo os participantes, os quatro fatores mais relevantes para a composição de times em *League of Legends* são: a estratégia, a habilidade dos jogadores com os campeões em questão, o meta e o *counterpicking*. Podendo escolher mais de uma opção, cada um teve 73, 57, 43 e 35 votos, respectivamente.

### 5.2.2 ESTRATÉGIA *POKE/PUSHER*

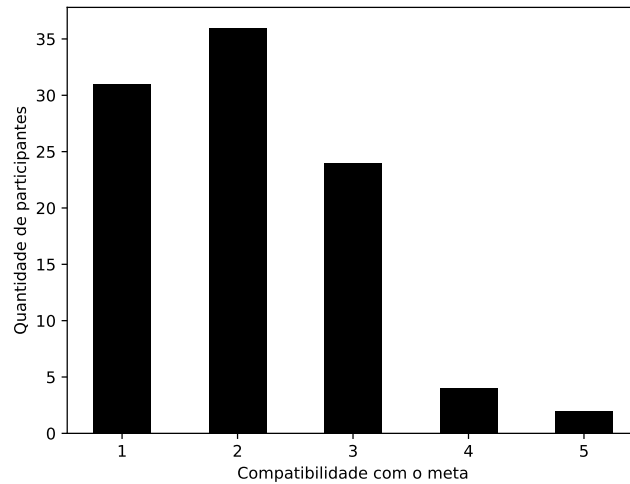
Quando foi perguntado sobre a compatibilidade desta composição com o meta, os participantes responderam conforme mostrado na Figura 16.

<sup>2</sup>Elo é o nome dado ao ranqueamento de jogadores dentro de uma partida competitiva.

<sup>3</sup>A escala Likert é um tipo de escala de resposta psicométrica utilizada em questionários. Ao responderem a um questionário baseado nesta escala, os perguntados especificam seu nível de concordância com uma afirmação.



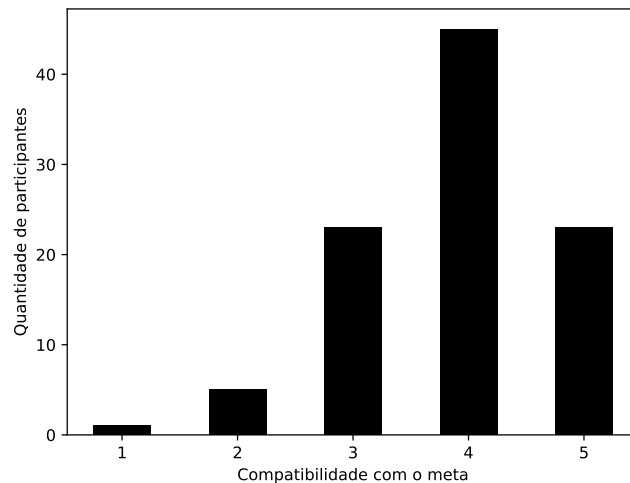
Figura 16: Compatibilidade da composição de *Poke* com o meta da versão 9.20



### 5.2.3 ESTRATÉGIA *HARD ENGAGE*

A Figura 17 mostra as respostas obtidas para a estratégia de *Hard Engage* e é possível notar o grande número de participantes que votaram na opção 4, tornando-a viável nesse contexto.

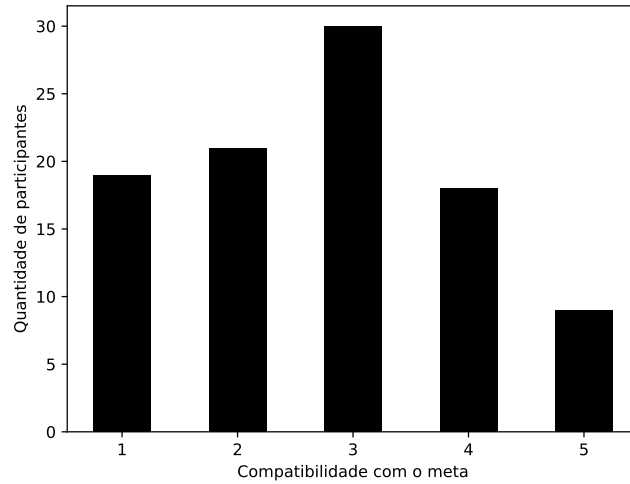
Figura 17: Compatibilidade da composição de *Hard Engage* com o meta da versão 9.20



### 5.2.4 ESTRATÉGIA *TEAM FIGHT*

Quando foi perguntado sobre a compatibilidade da composição gerada para a estratégia de *Team Fight*, os participantes responderam conforme exibido na Figura 18.

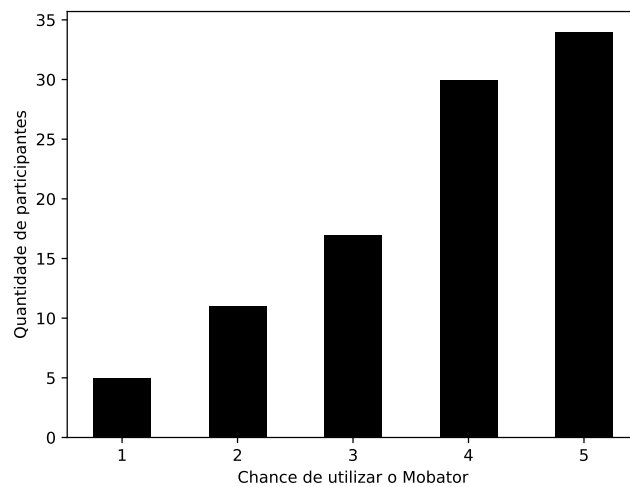
**Figura 18: Compatibilidade da composição de Team Fight com o meta da versão 9.20**



Por fim, na última seção do *survey* os participantes foram encorajados a expressar suas opiniões acerca do Mobator, e 12 participantes sugeriram modificações na abordagem.

Os participantes do questionário também se mostraram abertos para a utilização de abordagens e ferramentas inteligentes para a geração de composições em League, como podemos ver na Figura 19, onde 34 participantes demonstraram grandes chances de utilização de uma ferramenta inteligente.

**Figura 19: Change de utilizar o Mobator**



Após uma análise empírica das respostas, melhorias foram realizadas na função de *fitness* (deixando-a conforme apresentado na Seção 4.4) visando alcançar uma aceitação ainda maior por parte dos participantes do *survey*. Assim, um novo formulário foi criado

alterando as composições iniciais pelas novas obtidas após reexecução do algoritmo. Os novos resultados obtidos são apresentados na Subseção 5.2.5;

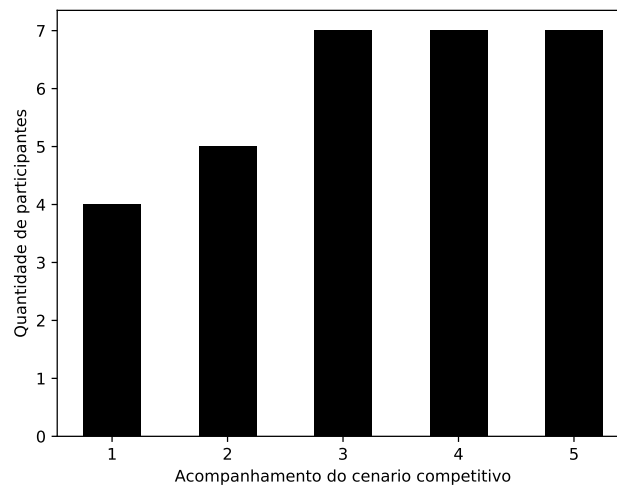
### 5.2.5 SURVEY VERSÃO II

Conforme citado anteriormente, não houveram mudanças estruturais no *survey* aplicado, somente nas composições que seriam avaliadas pelos participantes.

A versão II do questionário foi distribuída via e-mail para os participantes que haviam respondido a versão II do mesmo. Essa distribuição ocorreu entre 15 e 20 de novembro de 2019. Após este período, os dados coletados foram analisados, visando compreender a opinião dos jogadores à respeito das composições propostas pelo Mobator.

No total, foram recebidas 30 respostas de jogadores que, em sua maioria (66.7%), se consideram experientes e/ou competitivos. Essa informação pode ser confirmada pelo ranqueamento dos mesmos, já que 10 participantes possuem um dos 3 maiores elos do jogo. A frequência com que esses jogadores acompanham o cenário competitivo de League está demonstrada na Figura 20.

**Figura 20: Acompanhamento do cenário competitivo**

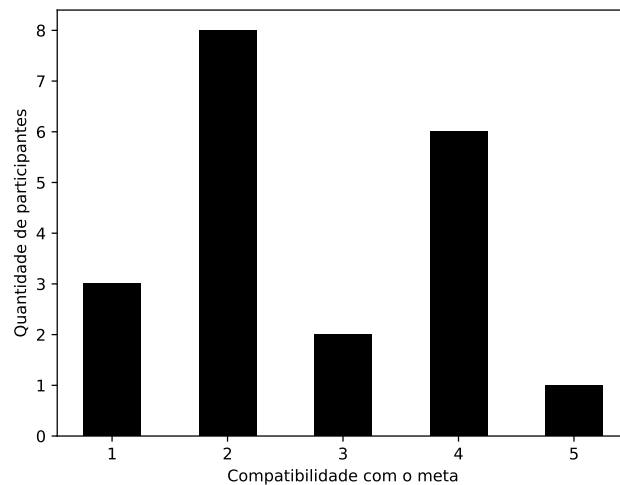


Assim como na primeira execução do formulário os parâmetros mais importantes para geração de composições foram: estratégias de composição, habilidades dos jogadores com os campeões, meta e *counterpicking*.

### 5.2.6 ESTRATÉGIA POKE/PUSHER

Os resultados da avaliação da estratégia de Poke estão disponíveis na Figura 21. Diferente da primeira execução do formulário, dessa vez as opiniões ficaram mais divididas entre as notas 2 e 4, mas ainda com uma opinião negativa sobre a composição em relação ao meta.

**Figura 21: Compatibilidade da composição de Poke com o meta do patch 9.20**



### 5.2.7 ESTRATÉGIA HARD ENGAGE

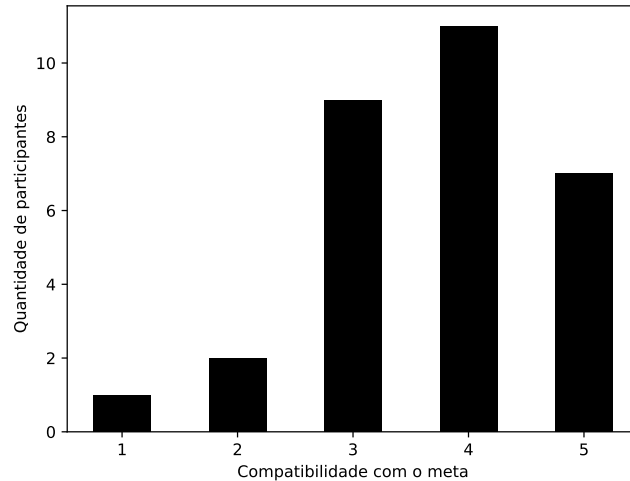
Quando questionados sobre a composição gerada com base na estratégia de Hard Engage, as respostas seguiram conforme mostra a Figura 22. Tem-se, novamente, uma boa aceitação da abordagem para esse contexto, visto que a maioria das respostas está no final da escala (com valores 4 e 5).

O resultado positivo em relação ao meta também fica claro na Figura 23, onde são exibidas as respostas da probabilidade dos jogadores utilizarem essa composição em uma partida de League.

### 5.2.8 ESTRATÉGIA TEAM FIGHT

Na Figura 24 estão os resultados obtidos acerca da composição gerada para a estratégia de Team Fight. Mais uma vez, as avaliações ficam, em sua maioria, nos valores 4 e 5 da escala, demonstrando concordância com o meta. Além disso, a maioria dos jogadores sinalizaram que utilizariam a composição em uma partida de League.

**Figura 22: Compatibilidade da composição de Hard Engage com o meta do patch 9.20**



### 5.3 RESULTADOS

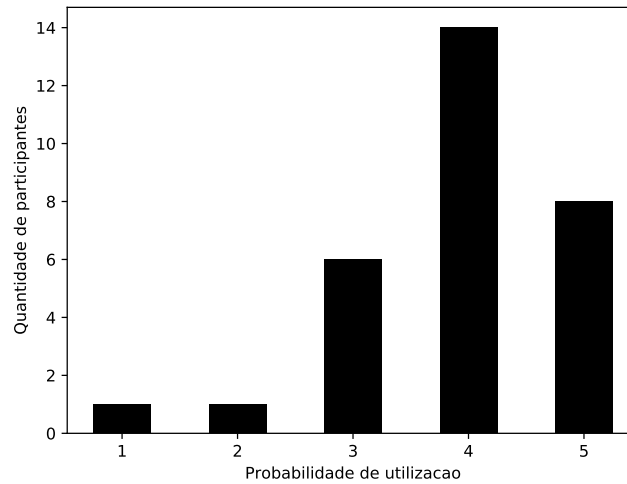
O experimento foi conduzido para analisar e avaliar a efetividade e eficiência da abordagem proposta para geração de equipes em League of Legends ao combinar atributos estatísticos, empíricos e estratégias de jogo. As orientações de Wohlin (2000) foram seguidas para estas execuções. Todos os experimentos foram realizados em um notebook com um CPU Intel Core i5 2.30GHz, 16GB de memória e um sistema operacional Debian GNU, e foram conduzidos da seguinte maneira:

1. Diferentes times ( $T$ ) foram gerados como objetos de estudo. Estes times variavam de acordo com as estratégias.
2. Os time foram avaliados de acordo com a função de *fitness* proposta. O valor obtido por meio da avaliação corresponde diretamente à qualidade da composição gerada.
3. Os valores de avaliação de cada time de uma estratégia foram computados.
4. A duração de cada execução foi armazenada em milissegundos para cada uma das três estratégias propostas.

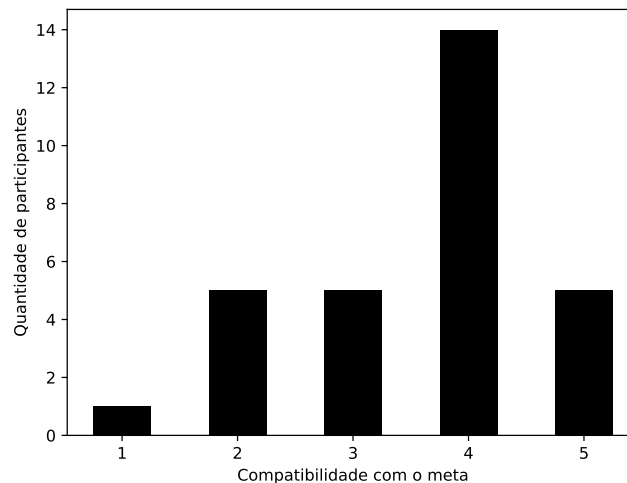
Os resultados dos experimentos serão exibidos nas próximas subseções de acordo com cada uma das Questões de Pesquisa.

**QP1: Para qual estratégia a abordagem para geração de composições mostra-se mais eficaz?**

**Figura 23: Probabilidade de utilização da composição**



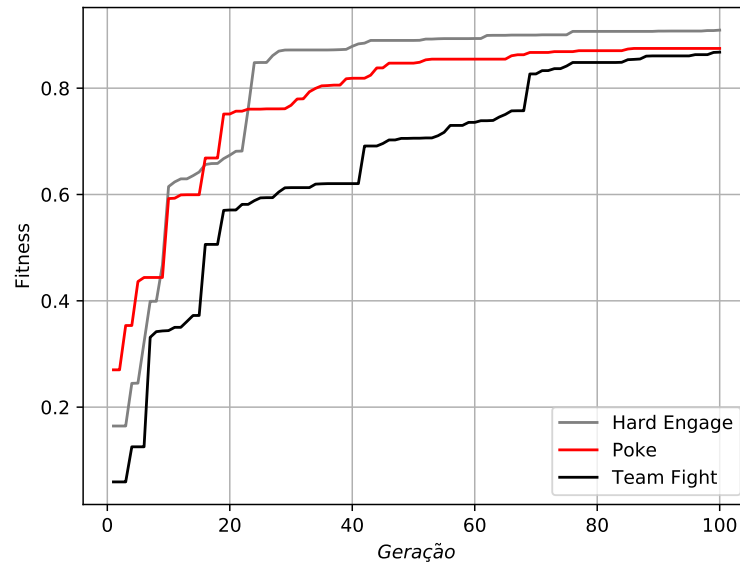
**Figura 24: Compatibilidade da composição de Team Fight com o meta do patch 9.20**



Na Figura 25 as linhas representam os valores de *fitness* obtidos em cada geração e estratégia utilizando a configuração 1. No eixo X temos as gerações do algoritmo e no eixo Y o valor de *fitness* obtido. O valor máximo de *fitness* é 1.0 pois o mesmo foi normalizado de acordo com as funções de avaliação, sendo assim o máximo que as estratégias podem alcançar é 100% disso. Os resultados indicam que a estratégia de Hard Engage, em média, alcançou o melhor valor de avaliação que foi 90.91%. Enquanto isso, a estratégia de Poke alcançou um valor de 87.47% e a avaliação de Team Fight ficou próxima de 86.77%.

Os resultados encontrados ao executar a configuração 2 são apresentados na Figura 26. Observa-se que o valor obtido por todas as estratégias é mais alto em comparação com a primeira configuração analisada. Portanto, é possível notar um aumento de 5.76% no

Figura 25: Valor de fitness alcançado utilizando a  $cfg_1$  para cada estratégia.



valor alcançado pela estratégia de Hard Engage, 8.62% para Poke e 7.5% para Team Fight.

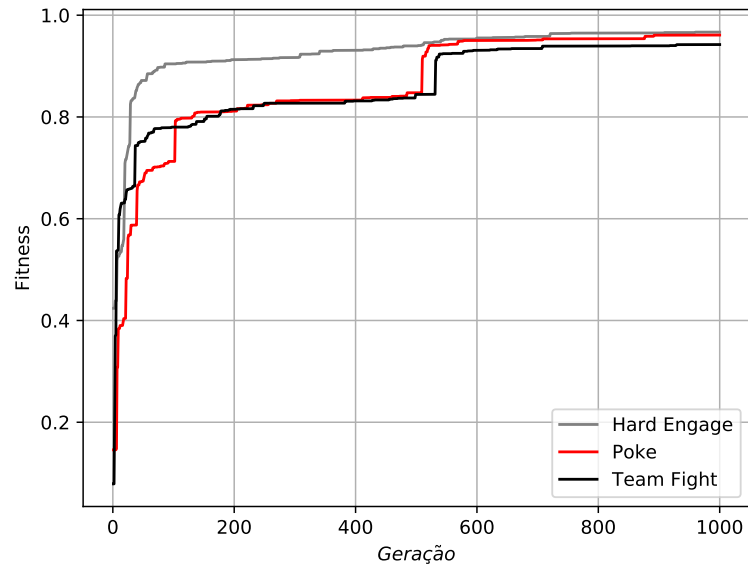
Ao executar os experimentos utilizando a configuração 3 foi alcançado o valor de 89.80% para a estratégia de Hard Engage. Poke alcançou uma avaliação de 88.01% e Team Fight 83.96%. Apesar do valor de Hard Engage ter sido menor do que o encontrado com a configuração 1, observa-se que as demais estratégias obtiveram avaliações melhores em comparação com esta mesma configuração.

Por fim, os valores obtidos ao utilizar a  $cfg_4$  são exibidos na Figura 27. Hard Engage alcançou um valor de 96.87%, Poke 96.88% e Team Fight 93.36%. É natural que esses resultados sejam os mais altos devido ao número elevado da população e de gerações, principalmente devido ao grande espaço de busca que este problema possui.

Para certificar a confiabilidade dos dados obtidos na  $(QP_1)$  foi realizado teste de hipótese. Como não se espera que as amostras utilizadas tenham distribuição normal, foi utilizado teste não paramétrico para verificar as hipóteses. Neste caso, foram comparados os resultados produzidos pela  $cfg_4$ , já que esta obteve os melhores valores de *fitness*. A hipótese nula ( $H_{10}$ ) a ser testada trata que as médias das amostras são iguais. Portanto, ( $H_{10}$ ) somente é rejeitada se o conjunto de requisitos produzir resultados significativos, reforçando a hipótese alternativa ( $H_{11}$ ) que defende que os dados analisados são estatisticamente diferentes.

Para avaliar a confiabilidade dos dados resultantes na questão  $(QP_1)$  foi aplicado o Teste de Kruskal Wallis (KW) com a finalidade de comparar a diferença das estratégias

Figura 26: Valor de fitness alcançado utilizando a  $cfg_2$  para cada estratégia.



estudadas. Este é um teste estatístico geralmente utilizado para verificar se existem diferenças significativas entre os resultados de 3 ou mais amostras. Na questão ( $QP_1$ ), a hipótese nula ( $H_{10}$ ) para o teste KW indica que não existe diferenças entre os valores de *fitness* para as diferentes estratégias. Portanto, se o teste KW produzir evidências sobre a diferença entre Poke, Hard Engage e Team Fight, ou seja, rejeitando a ( $H_{10}$ ), então a hipótese alternativa ( $H_{11}$ ) que considera diferença nos valores de *fitness* alcançados podem ser assumidos. O resultado do teste de KW é apresentado na Tabela 7.

Tabela 7: Resultados do teste de Kruskal Wallis

Atributo	Valor
Kruskal Wallis qui-quadrado	960,236
Graus de Liberdade	2
P-valor	0,000

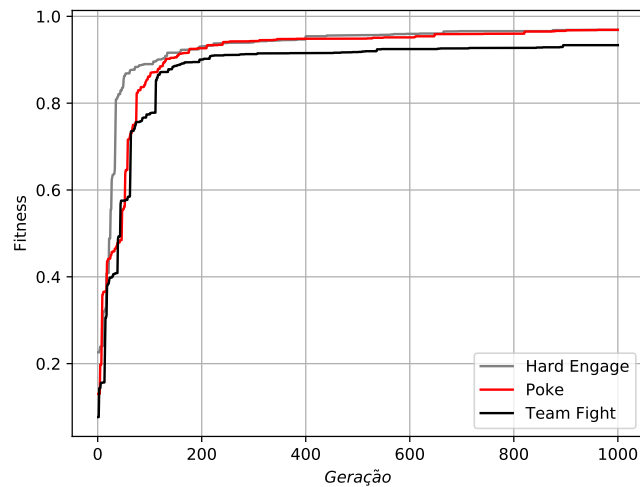
Fonte: Autoria própria

Conforme pode ser visto na Tabela 7, as funções de *fitness* obtiveram p-values inferiores a margem de erro (0,05), ou seja, mantiveram um nível de confiança de 95%. Assim, existem evidências das diferenças entre os valores resultantes da função de avaliação  $cfg_4$  para as três estratégias do estudo.

Os resultados assim obtidos são compatíveis com as expectativas das estratégias em um contexto matemático e empírico. Com  $cfg_4$ , os valores encontrados ficaram entre 93% e 96% e as composições poderiam ser utilizadas em uma partida de League, pois os



Figura 27: Valor de fitness alcançado utilizando a  $cfg_4$  para cada estratégia.



campeões demonstram sinergia entre si e se encaixam na estratégia proposta.

As Figuras 28, 29, 30 mostram composições geradas pelo Mobator utilizando estratégias de Hard Engage, Poke e Team Fight, respectivamente, utilizando a  $cfg_4$ .

Figura 28: Composição gerada utilizando a estratégia de Hard Engage.



Figura 29: Composição gerada utilizando a estratégia de Poke.



Além de possuírem um excelente valor de *fitness* e habilidades que são indispensáveis para as composições analisadas, as equipes geradas também respeitam a obrigatoriedade de rotas explicadas na Seção 2.1, possuindo um campeão para cada papel do jogo.

A abordagem proposta pode ser vista como uma opção viável para ser utilizada neste contexto visto que a maioria das abordagens e ferramentas utilizam predição para sugerir equipes. Além do mais, técnicas de busca com uma função de avaliação adequada fornecem um grande potencial para satisfazer até as restrições mais duras.

Conforme visto nos gráficos apresentados, a utilização de AG pode conduzir

Figura 30: Composição gerada utilizando a estratégia de Team Fight.



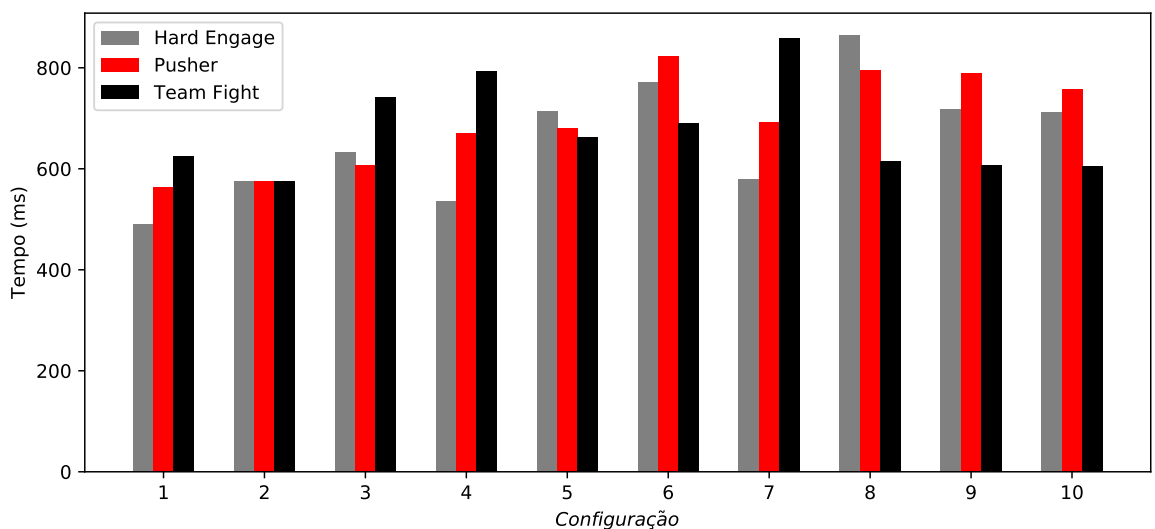
a busca para boas soluções, entretanto é necessário um número elevado de iterações para que os resultados sejam satisfatórios em um espaço de busca grande como o deste estudo. Por isso, faz-se necessária uma boa análise da relação custo-benefício entre o custo computacional para processar o algoritmo e o resultado encontrado.

**QP2: Qual estratégia é a mais eficiente para geração de composições?**

Para responder a ( $QP_2$ ), este segundo experimento armazenou a média de tempo para cada estratégia usando a melhor configuração ( $cfg_4$ ). A Figura 31 reporta os resultados obtidos baseando-se no tempo e no número de execuções.

Os resultados demonstram que a estratégia de Hard Engage, em média, obteve o tempo mais baixo para gerar composições: 659 milissegundos. Em contrapartida, a estratégia de Poke/Pusher foi a que consumiu mais tempo com uma média de 695 milissegundos. Considerando todas as estratégias, a primeira execução foi a mais rápida de todas com uma média de 559 milissegundos.

Figura 31: Média de tempo de acordo com  $cfg_4$  para cada estratégia.



A avaliação de confiabilidade dos dados resultantes na questão ( $QP_2$ ) também se deu através do teste de Kruskal Wallis, dessa vez com a intuito de comparar a diferença

de tempo de execução entre as funções estratégias Hard Engage, Team Fight e Poke com  $cfg_4$ . Na questão ( $QP_2$ ), a hipótese nula ( $H2_0$ ) para o teste KW indica que não existe diferenças entre o tempos de execução entre as estratégias. Portanto, se o teste produzir evidências sobre a diferença entre Hard Engage, Team Fight e Poke, a ( $H2_0$ ) é rejeitada, assumindo-se então a hipótese alternativa ( $H2_1$ ) que considera diferença entre o tempo de execução entre as estratégias. Na tabela 8 são apresentados os resultados do teste de Kruskal Wallis.

**Tabela 8: Resultados do teste de Kruskal Wallis para o contexto de tempo**

Atributo	Valor
Kruskal Wallis qui-quadrado	0,551
Graus de Liberdade	2
P-valor	0,759

**Fonte: Autoria própria**

Conforme pode-se notar na Tabela 8, as estratégias obtiveram p-valor acima da margem de erro (0,05). Portanto, a hipótese nula ( $H2_0$ ) não é descartada nesse caso, sendo assim é confirmado que não existe diferença de tempo de execução entre as estratégias utilizando  $cfg_4$ .

### **QP3: Qual é a melhor estratégia para geração de composições?**

Responde-se a ( $QP_3$ ) com base nas duas questões iniciais e também em ambos os *survey* realizados com os jogadores de League.

Após se destacar no valor de *fitness* alcançado, chegando a 96.87% na média de execuções utilizando  $cfg_4$ , a estratégia de Hard Engage também foi a mais aceita pelos participantes do survey, tanto no contexto de meta quanto pela possível utilização em uma partida, como observado nas QPs acima.

**Tabela 9: Comparação das estratégias com  $cfg_4$**

Estratégia	Maior <i>Fitness</i>	Média de tempo
Poke	96,88	695 ms
Hard Engage	96,87	659 ms
Team Fight	93,36	677 ms

**Fonte: Autoria própria**

Na Tabela 9 é apresentada uma comparação entre as diferentes funções de avaliação e as médias dos valores de *fitness* e de tempo (em milissegundos) alcançados. Este artefato também demonstra a superioridade dos resultados obtidos pela estratégia de Hard Engage,

já que ela possui o melhor *trade-off* (custo-benefício) por possuir uma média elevada com um baixo tempo de processamento.

Por ter um objetivo mais claro e bem definido, os campeões dessa estratégia possuem características bem específicas, normalmente ligadas a habilidades debilitantes, e isso se faz presente em todos eles. Esse fato pode fazer com que a convergência para um ponto otimizado no espaço de busca seja mais rápida.

## 6 CONSIDERAÇÕES FINAIS

Envolver a academia e a indústria é um dos grandes desafios das pesquisas e o contexto de jogos eletrônicos tem se mostrado um bom cenário para a aplicação dessa relação. A utilização de técnicas inteligentes para o contexto deste trabalho ainda é um tema em crescimento, principalmente baseando-se no cenário brasileiro, onde existem poucos estudos publicados envolvendo as áreas de pesquisa pertinentes a este trabalho. Esse déficit de trabalhos relacionados ao contexto, gera uma grande motivação para a execução desta pesquisa.

A abordagem proposta emprega um Algoritmo Genético conhecido como AG, o qual é guiado por um esquema de *fitness* para gerar uma equipe baseando-se em uma estratégia definida como parâmetro.

Nas demais seções deste capítulo, são apresentadas as contribuições e limitações da abordagem (Seção 6.1 e 6.2, respectivamente) e por fim, o capítulo se encerra na Seção 6.3, com uma breve discussão sobre os possíveis trabalhos futuros a serem abordados.

### 6.1 CONTRIBUIÇÕES

A principal contribuição desta pesquisa é prover uma abordagem inteligente para a geração de equipes em jogos MOBA. Mais especificamente, as contribuições podem ser sumarizadas como:

1. uma função de *fitness* para auxiliar a composição de equipes em diferentes jogos MOBA;
2. uma nova abordagem para a formação de equipes;
3. um estudo empírico para avaliação das composições produzidas pela abordagem, considerando os seguintes itens:

- um experimento individual para avaliar cada uma das três estratégias consideradas;  
e
- um questionário avaliativo dos resultados da abordagem;

## 6.2 LIMITAÇÕES

Devido a própria natureza dos jogos MOBA e suas composições e estratégias, algumas limitações podem ser identificadas na abordagem, principalmente pela complexidade do contexto de gerações de composições. Assim, esse projeto possui as seguintes limitações:

1. a base de dados precisa ser atualizada constantemente de acordo com as informações disponibilizadas pela Riot Games, já que essa é a única fonte confiável.
2. por se tratar de uma ferramenta de terceiros, as taxas de vitória obtidas no op.gg podem não ser precisas, fazendo com que a função de *fitness* seja afetada.
3. a validação realizada via *survey* necessita de um número considerável de participantes para que possa ser validada, e ainda sim não é completamente confiável.

## 6.3 TRABALHOS FUTUROS

Apesar dos resultados promissores, ainda há muitos desafios para serem superados no contexto de geração de equipes utilizando técnicas inteligentes. Para uma geração de equipes realmente otimizada e plausível, o time adversário também deveria ser analisado pelo algoritmo. Nesse sentido, um trabalho futuro envolve incluir os campeões inimigos como parâmetro para a utilização da técnica de *counterpicking*. Além disso, a sinergia entre os campeões da mesma equipe também pode ser considerada por uma abordagem futura.

Outro trabalho futuro seria analisar a taxa de vitória da composição final utilizando técnicas de aprendizagem de máquina, baseando-se em resultados de partidas passadas. Também seria interessante aplicar a abordagem proposta para outros jogos MOBA como Dota 2.

## REFERÊNCIAS

- BASILI, V.; WEISS, D. A methodology for collecting valid software engineering data. **IEEE Transactions on Software Engineering**, v. 10, n. 6, p. 728–738, 1986.
- CHENG, Z. et al. What makes a good team? A large-scale study on the effect of team composition in honor of kings. **CoRR**, 2019. Disponível em: <<http://arxiv.org/abs/1902.06432>>.
- CONLEY, K. E.; PERRY, D. How does he saw me? a recommendation engine for picking heroes in dota 2. In: . [S.l.: s.n.], 2013.
- COSTA, L. M.; SOUZA, F. C. M.; SOUZA, A. C. An approach for team composition in league of legends using genetic algorithm. In: . [S.l.: s.n.], 2019.
- DARWIN, C. **A origem das espécies**. [S.l.]: Edipro, 2018. 480 p.
- DEAVEN, D. M.; HO, K. M. Molecular geometry optimization with a genetic algorithm. **Phys. Rev. Lett.**, American Physical Society, v. 75, p. 288–291, 1995.
- EGGERT MARC HERRLICH, J. S. C.; MALAKA, R. Classification of Player Roles in the Team-Based Multi-player Game Dota 2. In: **14th International Conference on Entertainment Computing (ICEC)**. [s.n.], 2015. LNCS-9353, p. 112–125. Part 1: Full papers. Disponível em: <<https://hal.inria.fr/hal-01758447>>.
- FERRARI, S. From generative to conventional play: Moba and league of legends. In: **DiGRA Conference**. [S.l.: s.n.], 2013.
- HOLLAND, J. H. **Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence**. [S.l.]: MIT Press, 1992.
- JR, T. T. G.; MANN, E. R. **Cathode-ray tube amusement device**. [S.l.]: Google Patents, dez. 14 1948. US Patent 2,455,992.
- KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: **Proceedings of ICNN'95 - International Conference on Neural Networks**. [S.l.: s.n.], 1995. v. 4, p. 1942–1948.
- KIM, J. et al. The proficiency-congruency dilemma: Virtual team design and performance in multiplayer online games. In: . [s.n.], 2016. p. 4351–4365. Disponível em: <<http://doi.acm.org/10.1145/2858036.2858464>>.
- LOOI, W. et al. Recommender system for items in dota 2. **IEEE Transactions on Games**, 2018.
- LUQUE, L. Algoritmos genéticos em java. **JavaMagazine**, v. 82, 08 2010.

MORA-CANTALLOPS, M.; SICILIA, M.- Moba games: A literature review. **Entertainment Computing**, v. 26, 02 2018.

NASCIMENTO, F. A. F. do. **Modelagem biométrica e planejamento florestal otimizado utilizando a meta-heurística enxame de partículas**. 2010.

OLIVEIRA, V. da C. et al. A hybrid approach to build automatic team composition in league of legends. 2017.

PRODANOV, C.; FREITAS, C. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico**. [S.l.: s.n.], 2013.

RICH, E.; KNIGHT, K. **Artificial Intelligence**. [S.l.: s.n.], 2001.

ROGERS, S. Level up: um guia para o design de grandes jogos. **São Paulo: Blucher**, 2012.

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. [S.l.: s.n.], 2009. 73-82 p.

SALAS, E.; COOKE, N. J.; ROSEN, M. A. On teams, teamwork, and team performance: Discoveries and developments. **Human Factors**, v. 50, n. 3, p. 540–547, 2008. Disponível em: <<https://doi.org/10.1518/001872008X288457>>.

SAPIENZA, A.; GOYAL, P.; FERRARA, E. Deep neural networks for optimal team composition. 2018. Disponível em: <<http://arxiv.org/abs/1805.03285>>.

SOUZA, F. C. M. **Uma abordagem para geração de dados de teste para o teste de mutação utilizando técnicas baseadas em busca**. São Carlos, SP, Brasil: [s.n.], 2017.

SOUZA, R. T. de. Aplicação de algoritmos classificadores para previsão de vitória em uma partida de league of legends. 2017.

SUZNJEVIC, M.; MATIJASEVIC, M.; KONFIC, J. Application context based algorithm for player skill evaluation in moba games. In: . [S.l.: s.n.], 2015. p. 1–6.

TANUAR, E. et al. Back propagation neural network experiment on team matchmaking moba game. In: . [S.l.: s.n.], 2018. p. 240–243.

UBISOFT. **Ubisoft Q3 FY18**. São Paulo, 2018. Disponível em: <<http://bit.ly/2Jpts1E>>. Acesso em: 23/05/2019.

WANG, W. Predicting multiplayer online battle arena (moba) game outcome based on hero draft data. In: . [S.l.: s.n.], 2016.

WOHLIN, C. **Experimentation in Software Engineering: An Introduction**. Kluwer Academic, 2000. ISBN 9780792386827. Disponível em: <<https://books.google.com.br/books?id=nG2UShV0wAEC>>.

WOLFE, R. A. et al. Sport and organizational studies: Exploring synergy. **Journal of Management Inquiry**, v. 14, n. 2, p. 182–210, 2005. Disponível em: <<https://doi.org/10.1177/1056492605275245>>.



ZHAMRI, C. et al. A method for group formation using genetic algorithm. **International Journal on Computer Science and Engineering**, v. 02, p. 3060–3064, 01 2010.

APÊNDICE A - SURVEY VERSÃO I

# Uma abordagem inteligente para geração de times em jogos MOBA

Essa pesquisa será usada para o trabalho de conclusão de curso de Lincoln Magalhães Costa, aluno do curso de Engenharia de Software da Universidade Tecnológica Federal do Paraná (UTFPR) Campus Dois Vizinhos.

A proposta do trabalho é desenvolver uma ferramenta inteligente para auxiliar na geração de composições de equipes em League of Legends.

**\*Obrigatório**

1. **Endereço de e-mail \***

---

2. **Qual é o seu nick em League of Legends? \***

---

3. **Qual é o seu elo atual em League of Legends? Considere o maior. \***

---

4. **Em qual nível você acredita que o seu conhecimento e habilidades em League of Legends se encaixam? \***

*Marcar apenas uma oval.*

- Jogador novato/casual
- Jogador experiente/competitivo
- Jogador profissional

5. **Você acompanha o cenário competitivo de League? \***

*Marcar apenas uma oval.*

	1	2	3	4	5	
Não acompanho	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Acompanho muito

## Welcome to the League of Draven.

Se dependêssemos da opinião do Draven ele com certeza diria ser a parte mais importante de League.

Mas e o que você acha disso tudo?

**6. O que você considera mais importante na composição de times em League of Legends? \****Marque todas que se aplicam.*

- Meta
- Estratégias de composição
- Habilidades dos jogadores com os campeões que utilizarão
- Counterpicks
- Outro: \_\_\_\_\_

**Análise de composições: Poke/Pusher/Siege.****7. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \****Marcar apenas uma oval.*

1      2      3      4      5

---

Muito incompatível                  Muito compatível

**8. Quais as chances de você utilizar essa composição em uma partida? \****Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

**9. Justificativa das respostas acima.**


---



---



---



---



---

**Análise de composições: Hard Engage.**

10. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \*  
*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito incompatível                  Muito compatível

---

11. Quais as chances de você utilizar essa composição em uma partida? \*  
*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

---

12. Justificativa das respostas acima.

---

---

---

---

---

## Análise de composições: Team Fight.



13. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \*  
*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito incompatível                  Muito compatível

---

14. Quais as chances de você utilizar essa composição em uma partida? \*  
*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

---

**15. Justificativa das respostas acima.**

---

---

---

---

---

**I am evil, stop laughing!****16. Além das estratégias exibidas anteriormente, que outras você utiliza em suas partidas ou vê em campeonatos?**

---

**17. Quais as chances de você utilizar uma ferramenta inteligente para a geração de equipes em League of Legends? \***

*Marcar apenas uma oval.*

	1	2	3	4	5	
Muito baixas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito altas

---

**18. Você tem ideias, sugestões e propostas para contribuir com a pesquisa?**

---

**APÊNDICE B - SURVEY VERSÃO II**

# Uma abordagem inteligente para geração de times em jogos MOBA

Essa pesquisa será usada para o trabalho de conclusão de curso de Lincoln Magalhães Costa, aluno do curso de Engenharia de Software da Universidade Tecnológica Federal do Paraná (UTFPR) Campus Dois Vizinhos.

A proposta do trabalho é desenvolver uma ferramenta inteligente para auxiliar na geração de composições de equipes em League of Legends.

**\*Obrigatório**

1. **Endereço de e-mail \***

---

2. **Qual é o seu nick em League of Legends? \***

---

3. **Qual é o seu elo atual em League of Legends? Considere o maior e desconsidere o modo Teamfight Tactics. \***

---

4. **Em qual nível você acredita que o seu conhecimento e habilidades em League of Legends se encaixam? \***

*Marcar apenas uma oval.*

- Jogador novato/casual
- Jogador experiente/competitivo
- Jogador profissional/semi-profissional

5. **Você acompanha o cenário competitivo de League? \***

*Marcar apenas uma oval.*

1      2      3      4      5

---

Não acompanho                  Acompanho muito

---

## Welcome to the League of Draven.

Se dependêssemos da opinião do Draven ele com certeza diria ser a parte mais importante de League.

Mas e o que você acha disso tudo?



**6. O que você considera mais importante na composição de times em League of Legends? \***

*Marque todas que se aplicam.*

- Meta
- Estratégias de composição
- Habilidades dos jogadores com os campeões que utilizarão
- Counterpicks
- Outro: \_\_\_\_\_

**Análise de composições: Poke/Pusher.**



**7. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \***

*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito incompatível                  Muito compatível

**8. Quais as chances de você utilizar essa composição em uma partida? \***

*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

**9. Justificativa das respostas acima.**

---

---

---

---

---

**Análise de composições: Hard Engage.**



10. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \*

Marcar apenas uma oval.

1      2      3      4      5

---

Muito incompatível                  Muito compatível

11. Quais as chances de você utilizar essa composição em uma partida? \*

Marcar apenas uma oval.

1      2      3      4      5

---

Muito baixas                  Muito altas

12. Justificativa das respostas acima.

---

---

---

---

---

### Análise de composições: Teamfight.



13. Na sua opinião, o quão compatível com o meta atual (9.20) essa composição está? \*

Marcar apenas uma oval.

1      2      3      4      5

---

Muito incompatível                  Muito compatível

14. **Quais as chances de você utilizar essa composição em uma partida? \***

*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

---

15. **Justificativa das respostas acima.**

---

---

---

---

---

## I am evil, stop laughing!

16. **Além das estratégias exibidas anteriormente, que outras você utiliza em suas partidas ou vê em campeonatos?**

---

17. **Quais as chances de você utilizar uma ferramenta inteligente para a geração de equipes em League of Legends? \***

*Marcar apenas uma oval.*

1      2      3      4      5

---

Muito baixas                  Muito altas

---

18. **Você tem ideias, sugestões e propostas para contribuir com a pesquisa?**

---