



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS
CURSO DE LICENCIATURA EM COMPUTAÇÃO**

JOSÉ ALDO SILVA DA COSTA

**INTELIGÊNCIA ARTIFICIAL EM JOGOS DE TABULEIRO: PROPOSIÇÃO DE
UMA HEURÍSTICA PARA O JOGO DE DOMINÓS**

**PATOS - PB
2016**

JOSÉ ALDO SILVA DA COSTA

**INTELIGÊNCIA ARTIFICIAL EM JOGOS DE TABULEIRO: PROPOSIÇÃO DE
UMA HEURÍSTICA PARA O JOGO DE DOMINÓS**

Trabalho de Conclusão de Curso apresentado ao Curso de Licenciatura Plena em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Orientador: Prof. Dr. Wellington Candeia de Araújo.

**PATOS – PB
2016**

É expressamente proibida a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano da dissertação.

C838i Costa, José Aldo Silva da
Inteligência Artificial em jogos de tabuleiro [manuscrito] :
proposição de uma Heurística para o Jogo de Dominós / José Aldo
Silva da Costa. - 2016.
23 p. : il. color.

Digitado.
Trabalho de Conclusão de Curso (Graduação em Computação)
- Universidade Estadual da Paraíba, Centro de Ciências Exatas e
Sociais Aplicadas, 2016.
"Orientação: Prof. Dr. Wellington Candeia de Araújo,
CCEA".

1. Algoritmo Heurístico. 2. Dominós. 3. Aumento de
performance em Computação. I. Título.

21. ed. CDD 005

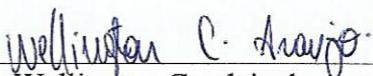
José Aldo Silva da Costa

**INTELIGÊNCIA ARTIFICIAL EM JOGOS DE TABULEIRO:
PROPOSIÇÃO DE UMA HEURÍSTICA PARA O JOGO DE DOMINÓS**

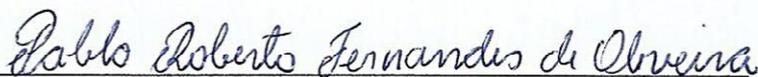
Trabalho de Conclusão de Curso apresentado ao
Curso de Licenciatura em Computação da
Universidade Estadual da Paraíba, em
cumprimento à exigência para obtenção do grau
de Licenciado em Computação

Aprovado em 23 de fevereiro de 2016

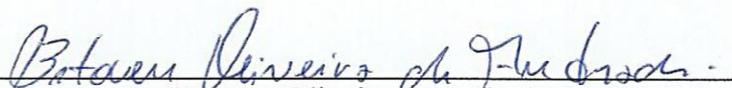
BANCA EXAMINADORA



Wellington Candeia de Araujo
(Orientador)



Pablo Roberto Fernandes de Oliveira
(Examinador)



Betoven Oliveira de Andrade
(Examinador)

INTELIGÊNCIA ARTIFICIAL EM JOGOS DE TABULEIRO: PROPOSIÇÃO DE UMA HEURÍSTICA PARA O JOGO DE DOMINÓS

José Aldo Silva da Costa¹

RESUMO

Este trabalho tem por objetivo analisar, desenvolver e aplicar técnicas de Inteligência Artificial a um dos jogos de tabuleiro mais conhecidos da história: Dominós. Desta forma, é proposto um algoritmo heurístico com a finalidade de aumentar a performance de um jogador de máquina ou um agente no ambiente do jogo. Para validar o algoritmo proposto, foram realizados testes que consistiram de simulações de partidas entre diversos agentes, incluindo um agente humano. Dados resultantes das excessivas interações ou competições entre os agentes mostraram-se bastante satisfatórios, apresentando um aumento de 45,77% no número de vitórias com o uso do algoritmo. Os dados são analisados e discutidos no decorrer deste trabalho.

Palavras-Chave: Algoritmo Heurístico. Dominós. Aumento de Performance.

1 INTRODUÇÃO

As aplicações de Inteligência Artificial (IA) em diversos campos do conhecimento têm sido objeto de estudo de diversos pesquisadores nas últimas décadas. Sem dúvida, o avanço destes estudos vem contribuindo bastante para o surgimento de novas técnicas que buscam proporcionar um maior desempenho em atividades diversas realizadas pelo ser humano.

Desde o início, a área de jogos tem se mostrado bastante favorável para essas aplicações. Buscando construir máquinas computacionais que sejam capazes de replicar habilidades cognitivas, tais como pensar e raciocinar, diversos pesquisadores têm estudado uma grande variedade de técnicas de aprendizagem e heurísticas visando uma melhoria no desempenho dessas máquinas no ambiente de jogos.

Com um dos trabalhos pioneiros na área, Shanon (1949) propõe um programa de computador que é capaz de jogar xadrez, impulsionando assim o desenvolvimento de pesquisas voltadas para o propósito da construção de máquinas que repliquem ou até superem as habilidades humanas em jogos. Tais pesquisas culminaram na derrota do campeão mundial Gary Kasparov em 1997 para Deep Blue, um computador dotado de inteligência artificial.

¹ Aluno de Graduação em Licenciatura em Computação na Universidade Estadual da Paraíba – Campus VII
Email: jsilvada@oswego.edu.br

De forma semelhante, outros jogos clássicos de tabuleiro como Gamão, Go, Othello, Damas entre outros ganharam também uma atenção especial e devido as suas características convidativas como diversão e desafio da mente, diversos algoritmos baseados em diferentes técnicas de IA são aplicados neste contexto.

Buscando investigar e contribuir para aplicações de IA em Jogos, este trabalho tem por objetivo analisar, desenvolver e aplicar técnicas de IA em dominós, um jogo de tabuleiro muito popular e que requer mais do que sorte dos seus jogadores, requer também habilidades estratégicas.

2 JOGO DE DOMINÓS

2.1 História do Jogo

A origem do jogo de dominós é bastante controversa. Alguns historiadores atribuem sua invenção aos chineses com o herói Hung Ming por volta de 234 a 181 A.C., na intenção de distrair seus soldados. Outra versão atribui sua invenção a um funcionário do imperador Hui Tsung no século XII (CELKO, 2001). O jogo apareceu na Itália em meados do século XVIII e espalhou-se pela Europa. Ainda para Celko (2001), apesar de as peças de dominós serem claramente uma herança chinesa, não existe nenhum debate sobre a possibilidade de essas serem as mesmas peças que poderiam ter sido levadas para Europa no século XIV ou se estas foram inventadas independentemente. Ainda no século XVIII, o jogo foi levado para França onde se tornou novidade.

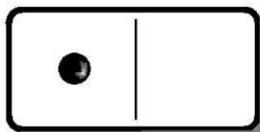
Em chinês, o jogo era conhecido como kua-p'ai ou "tabletes de osso". Acredita-se que o nome do jogo, tal como o conhecemos, vem da expressão francesa domino que por sua vez deriva do Latim dominus "senhor", uma forma de tratar com respeito os religiosos. O termo designava um traje com um capuz preto usado para ocultar a identidade de alguém. O jogo tornou-se primeiramente um sucesso entre os religiosos que ao completarem suas combinações requeridas exclamavam: *Benedicamus Domino* "louvemos ao Senhor" (SILVA, 2014).

Com o passar do tempo, devido a sua grande popularidade, o jogo ganhou diversas variações sendo influenciado de diversas maneiras por diferentes culturas. Possuindo características atrativas e desafiantes que envolviam raciocínio lógico e também o elemento da sorte, tais variações foram limitadas apenas pela criatividade de seus jogadores ao longo da história.

2.2 Características e Jogabilidade

Por se tratar de um jogo antigo, existem diversas variações atualmente. A versão mais popular do jogo de dominós presentemente no ocidente é o duplo-seis. É composto de 28 peças, também conhecidas por “pedras”, e cada uma delas é dividida por uma linha marcada ao meio que separa suas faces em duas partes ou “pontas”, como pode ser visto na Figura 1. Cada ponta possui um número que é expresso pela quantidade de pontos marcados que vai de 0 a 6.

Figura 1: Ilustração de uma pedra do jogo.

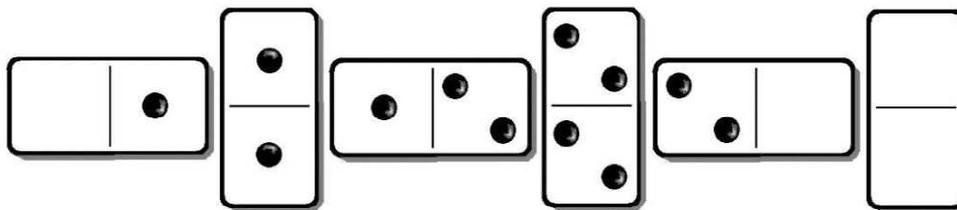


Fonte: Elaborada pelo autor

O número ideal de jogadores não é pré-definido podendo variar de acordo com a versão do jogo. Depois de misturar todas as pedras com faces viradas para baixo, cada jogador recebe aleatoriamente a mesma quantidade no início do jogo. Uma característica importante deste jogo é que os jogadores não têm conhecimento das pedras que seus oponentes possuem.

Pedras que possuem números iguais em ambas as pontas são denominados “duplos” ou “carroças”. Aquele que detém a maior carroça pode jogar primeiro. Se nenhum dos jogadores receber qualquer carroça então aquele que detém a maior pedra pode jogar primeiro. A “maior” pedra é determinada pela soma de pontos em suas pontas. Cada jogador adiciona uma pedra em uma das pontas do jogo, formadas pelas extremidades da cadeia que aos poucos vai se formando no decorrer da partida. Esta cadeia consiste de pedras adicionadas consecutivamente pelos jogadores, à medida que elas se encaixam, o que pode ser melhor entendido pela Figura 2.

Figura 2: Ilustração de uma cadeia de pedras do jogo.



Fonte: Elaborada pelo autor

Algumas versões utilizam uma carroça como cruzado ou spinner, e permitem aos jogadores utilizá-la como uma pedra com quatro pontas dando a ideia de duas cadeias que se cruzam (CELKO, 2001). Se jogador não tem a opção de adicionar qualquer pedra, por não possuir nenhuma que encaixe em qualquer das pontas no jogo, deve ir para o “cemitério”, como é denominado o restante de pedras que não foram distribuídas, e continuar a “roubar” uma de cada vez até encontrar alguma que encaixe em qualquer uma das pontas do jogo. O cemitério é formado por pedras restantes que não foram distribuídas no início do jogo.

O jogo termina quando um jogador consegue posicionar todas as suas pedras antes de todos os seus adversários ou quando a cadeia estiver “bloqueada” ou “fechada”, o que acontece quando não há mais nenhum movimento possível para qualquer um dos jogadores. Neste caso, o vencedor é aquele que tem a “menor” pedra na mão ou aquele com menor quantidade de pontos em todas as pedras na mão.

2.3 Regras da versão implementada

Este trabalho tem como foco a variação duplo-seis, que por sua vez detém também outras versões variantes, tais como Todos os Cinco ou Muggins, Bergen, Dominó Clássico, Dominó Clássico com spinners, Dominós com roubo, Dominós com Parceria, Dominós às cegas, Duplo-Bergen e outros (MCLEOD, 2015).

A variação escolhida como objeto de estudo implementada neste trabalho baseia-se principalmente na versão Dominó Clássico utilizando-se alguns elementos da versão Dominós com Roubo, e uma forma mais simples para estabelecer a pontuação. As regras presentes do dominó clássico utilizadas na versão implementada são descritas da seguinte forma:

- 2 jogadores recebem 10 pedras aleatórias cada um.
- Quem recebe a maior carroça começa, caso contrário, quem recebe a maior pedra. Se ambos não ocorrerem, as pedras são misturadas novamente e reinicia-se a partida.
- Cada jogador adiciona uma pedra na extremidade do jogo em que ela encaixe.
- O jogo termina quando um dos jogadores consegue encaixar todas as suas pedras, ou quando o jogo estiver bloqueado, ou seja, não é possível adicionar mais nenhuma pedra.

Os elementos da versão de dominó com roubo adicionadas às regras anteriores dizem respeito principalmente ao restante das pedras que não foram distribuídas, o “cemitério”. Na versão clássica todas as pedras são distribuídas e não existe cemitério. A versão com roubo contribui com o seguinte elemento para a versão implementada:

- Se o jogador não possui uma pedra que encaixe, ele deve se dirigir ao cemitério e “roubar” aleatoriamente uma das pedras que estão viradas com face para baixo enquanto houver. Se ela não encaixar, ele deve ficar com a pedra e repetir esta ação até que encontre alguma que encaixe ou não restar mais nenhuma no cemitério.

Algumas versões determinam o vencedor através da pontuação, levando em consideração valores associados à maneira como o jogador encaixa as pedras ao longo do jogo. Na maioria dos casos, a pontuação é cumulativa e o vencedor é o jogador que acumula mais pontos ao longo de várias partidas. Desta forma, para determinar o vencedor na versão implementada, mais um elemento foi adicionado:

- Uma vitória é determinada quando o jogo termina. Aquele que acumula mais vitórias durante um número determinado de partidas será o vencedor.

3 INTELIGÊNCIA ARTIFICIAL EM JOGOS

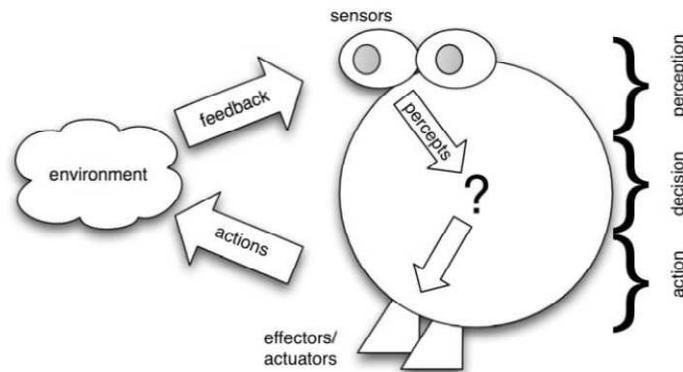
O contexto de jogos tem se mostrado um ambiente bastante favorável para aplicação de diversas técnicas de Inteligência Artificial. Habilidades cognitivas como raciocinar e aprender através de experiências, muito necessárias no contexto de jogo, parecia algo intrínseco apenas aos seres humanos até o advento do surgimento do campo estudos de IA. A partir de então, grande esforço vem sendo percebido com o principal intuito de replicar tais comportamentos e habilidades em máquinas ou programas de computador, característica esta que Winston (1992) considera essenciais para definir este campo de conhecimento.

Uma das melhores formas de estudar o comportamento de uma máquina dotada de estratégias é através da modelagem de agentes inteligentes em um contexto de jogos que exigem habilidades como raciocínio.

3.1 Agentes

Russel e Norvig (2003) definem um agente como qualquer entidade capaz de perceber seu mundo exterior ou seu ambiente através de sensores e agir neste ambiente através de atuadores. Tomando o ser humano como exemplo de um agente, os olhos e os ouvidos servem como sensores, o que utilizamos para perceber o mundo, que é o ambiente em que estamos inseridos. As pernas e mãos servem como atuadores, que utilizamos para atuar e modificar o ambiente a nossa volta. A principal particularidade que diferenciam agentes está na capacidade de tomar decisões. A descrição de uma agente genérico pode ser vista na Figura 3.

Figura 3: Descrição de um agente genérico



Fonte: JEFFREY, 2002

Ao definir um agente, Maes (1996) ressalta o conceito de um sistema computacional que habita um dado ambiente, sente e age nesse ambiente autonomamente, e ao fazê-lo realiza um conjunto de objetivos ou tarefas para os quais foi projetado. O cumprimento de objetivos é um fator determinante quando se pretender compreender como um agente atua e o que o leva a atuar de determinado modo no seu ambiente.

Wooldridge e Jennings (1996) definem um agente como uma peça de hardware ou, mais usualmente, um sistema computacional baseado em software que apresenta algumas propriedades, dentre elas:

- Autonomia – atuação com um grau de independência, sem a intervenção humana.
- Reatividade – ao perceber o ambiente, o agente responde rapidamente às alterações nele ocorridas.
- Pró-atividade - um comportamento direcionado a objetivos, não agindo somente em resposta às alterações no ambiente, mas sim, de acordo com um propósito.
- Habilidade Social – a capacidade a interação com outros agentes ou até pessoas, no momento adequado, visando concluir suas tarefas ou ajudar outros agentes.
- Racionalidade – o agente age de forma a atingir seus objetivos e não age de forma a impedir que esses mesmo objetivos sejam atingidos e para isso, sempre tomará a melhor decisão possível dadas suas circunstâncias.
- Inteligência – o estado do agente é formalizado pelo conhecimento como crenças, objetivos, planos, entre outros. Evidencia-se através da capacidade de raciocínio abstrato, resolver novos problemas ou adaptar-se à novas situações.

- Capacidade de aprendizagem – capacidade por vezes confundida com inteligência, esta diz respeito a habilidade do agente de adquirir novos conhecimentos e alterar o seu comportamento baseado em experiências prévias.

O fator da inteligência em agentes se torna bastante decisivo e complexo. Esta se relaciona principalmente com o fator da autonomia. Esse grau de autonomia ou independência pode variar de acordo com o objetivo para o qual os agentes são projetados.

Agentes inteligentes são entidades de software que realizam um conjunto de operações em nome de um dado utilizador ou de outro programa com um dado grau de independência ou autonomia, e ao fazê-lo, utilizam algum conhecimento ou representações dos objetivos e desejos do utilizador (IBM, 1997).

Russel e Norvig (1995) ainda destacam o fator da racionalidade, em que um agente inteligente deve optar, em um caso ideal, pela melhor ação possível em uma determinada situação. Por conseguinte, ele observa o seu ambiente e gera uma decisão baseada em um estudo racional que acontece anteriormente à sua ação.

Para a aplicação de IA em jogos, através da modelagem dos jogadores em um programa, a jogabilidade executada pelo computador se trata de uma simulação de agente que possui uma base de dados: O conhecimento de movimentos e jogadas clássicas, certos comportamentos que orientam suas decisões e que imitam os seres humanos, algoritmos de estratégias, registros de jogadas anteriores e base de previsão de jogadas, entre outros.

O ambiente no qual o agente está inserido se torna um fator decisivo que influencia na conquista do seu objetivo. Russel e Norvig (1995) ao abordarem este tópico, definem os ambientes de agentes como:

- Acessível vs. Inacessível – no ambiente acessível, a agente tem acesso a todos os estados do ambiente ou informação completa acerca dele.
- Determinístico vs. Estocástico – no determinístico, o próximo estado do ambiente é determinado pelo estado atual.
- Episódico vs. Sequencial – no episódico, a qualidade da ação do agente no ambiente depende apenas de si mesma.
- Estático vs. Dinâmico – Se o ambiente não mudar enquanto o agente age sobre ele então pode ser tido como estático, caso contrário, dinâmico.
- Discreto vs. Contínuo – no ambiente discreto, há um número limitado definido de percepções e ações do agente no ambiente.

Em um jogo de xadrez, por exemplo, o agente se depara com um ambiente caracterizado por informação completa ou acessível, determinístico, sequencial, estático e discreto. O pior caso para um agente seria um ambiente inacessível, sequencial, estocástico, dinâmico e contínuo, o que parece se aproximar de situações do mundo real.

3.2 Uma abordagem heurística

Problemas com alto grau de complexidade ou falta de acesso a um conjunto suficiente de informações necessárias sobre um determinado problema têm dificultado e até impossibilitado o encontro de soluções ótimas. Neste cenário, o emprego de métodos heurísticos na busca de soluções satisfatórias se torna ideal de acordo com Simon (1970).

O termo “heurística” vem do grego, que significa “descubro”, e pode ser entendido simplesmente como um conjunto de regras e métodos que conduzem à descoberta, à invenção e à resolução de problemas. Assim sendo, os métodos heurísticos podem ser compreendidos como algoritmos exploratórios cuja finalidade é resolver problemas. De acordo com Simon e Newell (1958), essa descoberta acontece de forma sequencial, alcançada através de sucessivas decisões, e essa ideia foi proposta baseada em suas observações de como as pessoas realizam várias escolhas sequenciais que acabam construindo a solução do problema.

Os métodos heurísticos nem sempre buscam encontrar uma solução ótima, exata ou definitiva para o problema, mas as melhores soluções possíveis a partir de sucessivas aproximações em direção a uma solução ótima. Para Bueno (2009), esta subjetividade ou falta de precisão não se trata de uma deficiência dos métodos heurísticos, mas uma particularidade análoga à inteligência humana, que busca resolver diversos problemas sem, contudo, conhecê-los com precisão. Assim, englobam estratégias, procedimentos, métodos de aproximação tentativa/erro, sempre na busca da melhor maneira de chegar a uma determinada solução.

Os processos heurísticos exigem muitas vezes menos tempo que os processos algorítmicos, aproximando-se mais da forma como o ser humano raciocina e chega às resoluções dos problemas, e garantem soluções eficientes.

4 METODOLOGIA E DESENVOLVIMENTO

O jogo de dominós foi implementado de forma a proporcionar a competição entre cinco competidores modelados: aleatório, ofensivo, defensivo, ofensivo e defensivo e um jogador humano por meio de entradas do teclado. O algoritmo foi implementado através da linguagem de programação CLISP (Common List Processing) fazendo uso da IDE Lispbox.

4.1 Modelagem do agente randômico

Dada a importância crucial de entender e implementar os movimentos particulares do jogo de forma adequada, a primeira fase do projeto concentrou-se na aprendizagem dos mesmos para uma possível implementação de acordo com as regras especificadas no contexto do jogo. Muito esforço foi feito para analisar a aplicabilidade dos movimentos permitidos no jogo sob diversas situações dadas.

Dois agentes com capacidade de realizar movimentos de forma aleatória foram modelados de forma que eles possam disputar uma partida um contra o outro e desta forma validar as regras de movimentos implementadas. Por movimentos de forma aleatória entende-se que os jogadores modelados tentam posicionar a primeira pedra que eles encontrarem que encaixem no jogo. Alguns destes movimentos podem ser observados na Figura 4.

Figura 4: Simulação de jogo: agentes randômicos

```
// Jose Aldo Silva da Costa
// Demo of the phase 1: Making moves and move applicability

CL-USER> (play)
PLAYER 1: ((0 3) (1 5) (6 6) (1 2) (0 1) (0 2) (2 5))
PLAYER 2: ((3 4) (0 4) (4 5) (1 6) (2 3) (3 5) (0 5))

STARTING WITH PLAYER 1
HAND: ((0 3) (1 5) (6 6) (1 2) (0 1) (0 2) (2 5))
MOVE: ((6 6))
HAND: ((0 3) (1 5) (1 2) (0 1) (0 2) (2 5))

CHAIN: ((6 6))

PLAYER 2
HAND: ((3 4) (0 4) (4 5) (1 6) (2 3) (3 5) (0 5))
MOVE: (1 6)

CHAIN: ((1 6) (6 6))
```

Fonte: Elaborada pelo autor

Os agentes continuam realizando jogadas enquanto a condição gameover for falsa. Esta condição é verificada no fim de cada jogada e se ela torna verdadeira, um dos agentes posicionou todas as suas pedras primeiro ou o jogo foi bloqueado. Por questões de limitações de tempo, optou-se por implementar uma forma mais simples, porém intuitiva, para exibição da cadeia gerada. A Figura 5 exemplifica como a cadeia exibindo a mensagem “Jogador 1 posicionou todas as pedras primeiro”.

Figura 5: Simulação de jogo: agentes randômicos

```
CHAIN
[1-0] [0-0] [0-6] [6-1] [1-1] [1-3] [3-0] [0-5] [5-5] [5-1] [1-2] [2-0] [0-4] [4-4] [4-5] [5-3] [3-3] [3-4] [4-2] [2-5]

W
Player 1 wins running out of tiles first!

NIL
```

Fonte: Elaborada pelo autor

4.2 Modelagem de um agente humano

A modelagem deste agente tornou-se necessária para permitir que um ser humano pudesse interagir com outros agentes implementados através de entradas pelo teclado. A amostra de um movimento efetuado por uma agente humano é demonstrada na Figura 6. Este agente tem a opção de escolher uma pedra de sua mão, que é notada entre parênteses, ou ir para o cemitério à procura de alguma que encaixe no jogo pressionando a letra “b”.

Figura 6: Simulação de jogo: Agente randômico vs. Agente humano

```
[5-5]

PLAYER 2
Please select a move from your hand ((0 6) (3 5) (2 6) (0 2) (3 3) (1 1) (2 4) (1 4) (4 6) (0 0)) or (press
b for Bone-yard):
(3 5)

[3-5][5-5]
```

Fonte: Elaborada pelo autor

O exemplo da Figura 7 mostra o agente recorrendo ao cemitério. De acordo com as regras definidas do jogo, os agentes só são permitidos ir para o cemitério quando eles não possuem nenhuma pedra que encaixe em qualquer uma das extremidades da cadeia.

Figura 7: Simulação de jogo: Agente humano vs. Agente humano

```
[4-4] [4-3] [3-5] [5-5] [5-2] [2-4] [4-0] [0-6] [6-1] [1-4] [4-6] [6-3] [3-3] [3-1] [1-1] [1-2] [2-6]

Please select a move from your hand ((0 2) (0 0) (2 2)) or (press b for Bone-yard):
b
You got a random tile: (5 6)
```

Fonte: Elaborada pelo autor

4.3 Modelagem de um agente heurístico

Programas computacionais no ambiente de jogos como xadrez, damas e Go envolvem informação completa, o que significa que o status completo do jogo é conhecido por cada jogador em todo o tempo, um ambiente de informação acessível. Em muitos casos, este aspecto permite que um elevado desempenho seja alcançado através de técnicas de inteligência artificial, tais como árvores de decisões.

No entanto, no jogo de dominó, nem toda a informação é conhecida ou acessível por todos os jogadores em todo o tempo, o que o caracteriza como um ambiente de informação incompleta ou inacessível. Um jogador não conhece as pedras do seu oponente ou as pedras do cemitério. Em um jogo com dois jogadores que possuem sete pedras cada um e com cemitério, apenas $\frac{1}{4}$ da informação do jogo é conhecido pelos jogadores no inicialmente.

Isto implica que ideia de árvores de busca pode não ser muito adequada para a implementação do algoritmo. De acordo com Simon (1970), a falta de acesso a informação impossibilita que tomadores de decisão encontrem soluções ótimas, todavia, ainda assim é possível chegar a soluções satisfatórias empregando heurísticas para tal.

Walklate (2007) concorda com a ideia de que o jogo de dominós é muito grande para ser modelado em forma de uma árvore completa de busca, sendo assim, métodos heurísticos apresentam um papel importante na tomada de decisão.

Alguns jogos são modelados de forma que diferentes jogadores são capazes de usar diferentes estratégias, como no jogo de Poker (DINIS, 2008). No caso particular deste trabalho, foram modelados três agentes heurísticos com estratégias diferentes. O primeiro jogador é capaz de utilizar uma abordagem mais defensiva, o segundo uma abordagem mais ofensiva e o último combina as estratégias ofensivas e defensivas formando uma abordagem híbrida.

Diversas estratégias foram apresentadas por pesquisadores como Celko (2001) e McLoad (2015) ao estudarem diversas versões do jogo. Quando seguidas corretamente, estas estratégias podem definir um jogador como sendo mais ofensivo e agressivo ou um jogador defensivo.

Por defensivo, entende-se que o jogador deve:

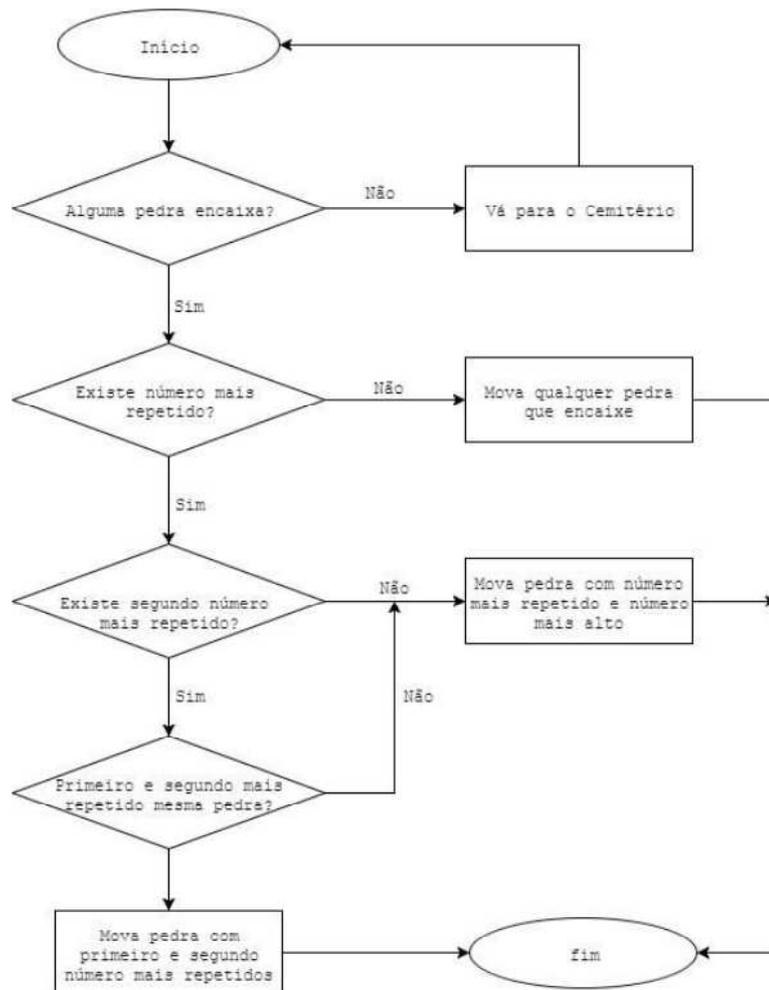
- Posicionar as carroças primeiro. Elas têm o mesmo número em ambos os lados, limitando opções de jogadas.
- Manter uma variedade de pedras pelo máximo de tempo possível
- Posicionar as pedras mais altas primeiro. No caso de a cadeia ser bloqueada, o jogador terá uma mão leve.

Por ofensivo, entende-se que o jogador deve:

- Perceber os pontos fracos do oponente lembrando dos naipes que ele não possui.
- Encontrar a extremidade mais vulnerável através da contagem dos naipes das próprias pedras mais as da cadeia.
- Tornar as extremidades da cadeia iguais, especialmente se foram um ponto fraco do oponente. O adversário terá menos oportunidades de posicionar suas pedras.

A estratégia fundamental do jogador defensivo é tentar manter uma variedade de números o maior tempo possível. Diferentes números possibilitam ao jogador estar preparado para eventuais jogadas ofensivas, principalmente bloqueios. As ações que desempenham papel significativo nessa abordagem são expressas no fluxograma da Figura 8. Este conjunto de ações é referente apenas à uma jogada, passando a sua vez para o adversário ao chegar em “fim”.

Figura 8: Fluxograma de uma jogada defensiva

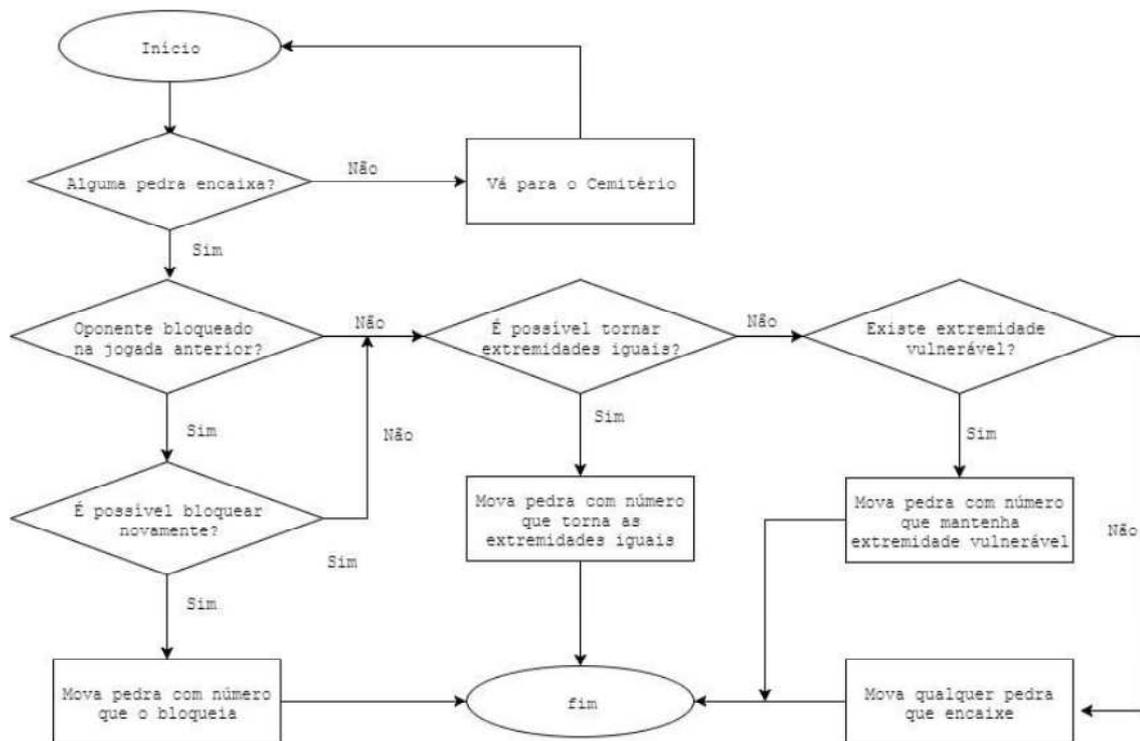


Fonte: Elaborada pelo autor

A estratégia fundamental do jogador ofensivo é tentar bloquear as jogadas de seu adversário. A estrutura de decisões do jogador ofensivo é ilustrado no fluxograma da Figura 9. Este jogador foi modelado de forma a possuir a habilidade de armazenar temporariamente a última jogada do seu oponente. Assim, a primeira ação a ser tomada por ele é verificar se houve um bloqueio, e se houve, ele tentará usar a mesma estratégia efetuando uma jogada similar.

Caso o outro jogador não tenha sido bloqueado, ele tentará diminuir as chances de jogada do seu oponente, reduzindo-as pela metade ao tornar as extremidades da cadeia iguais, ou seja, deixá-las com o mesmo número. Ainda se isso não for possível, ele tentará descobrir se existe alguma vulnerabilidade nas extremidades da cadeia do jogo ao contar os números repetidos. Quando nenhuma das condições for satisfeita, uma pedra aleatória é posicionada.

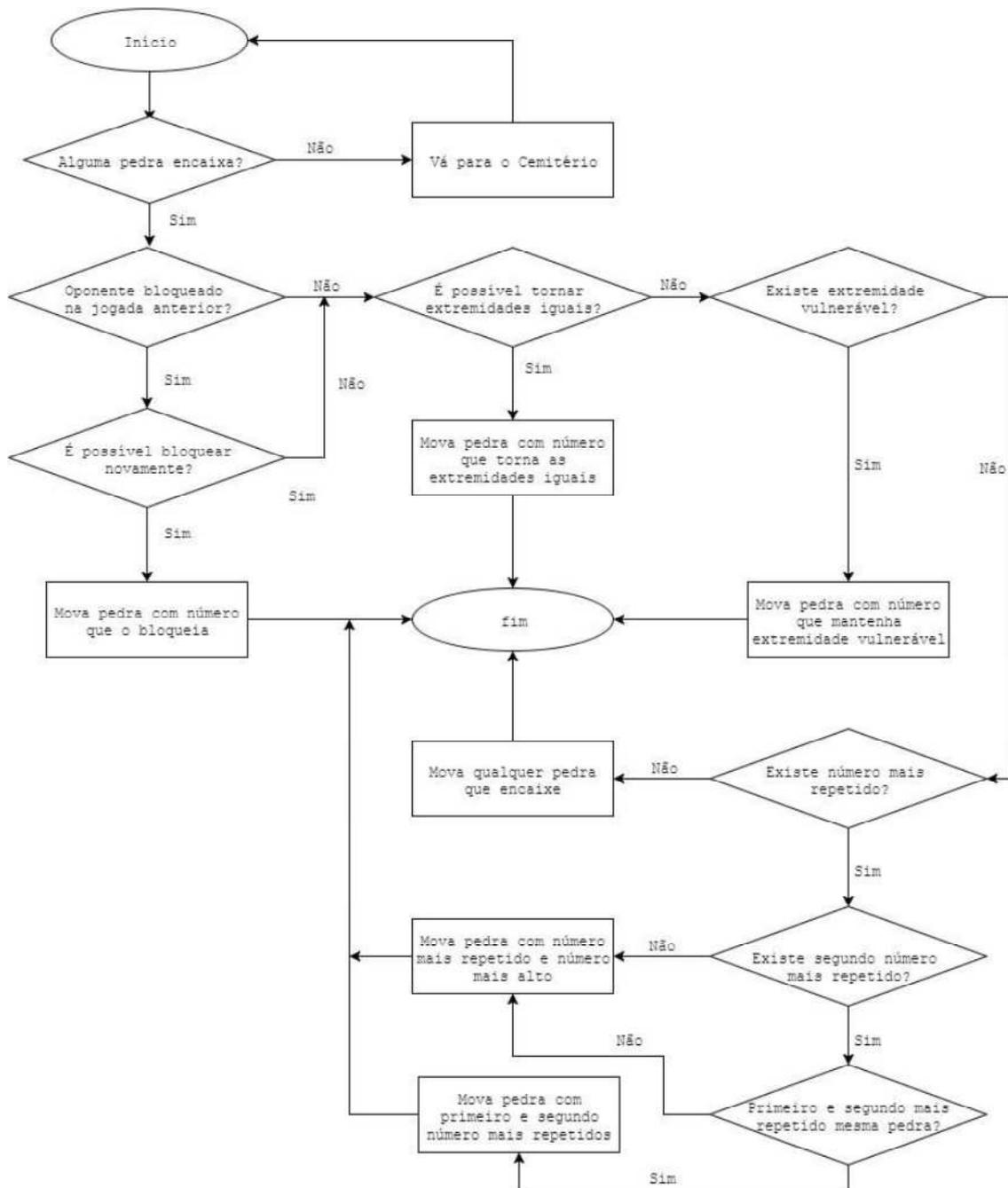
Figura 9: Fluxograma de uma jogada ofensiva



Fonte: Elaborada pelo autor

O agente heurístico ofensivo e defensivo é uma abordagem híbrida que combina jogadas ofensivas e defensivas. Em vez de selecionar uma pedra aleatoriamente quando nenhuma das condições para o jogador ofensivo for satisfeita, este jogador recorre às ações defensivas excluindo o fator de aleatoriedade usado pelo jogador ofensivo. Isto pode ser observado no fluxograma a seguir da Figura 10. O jogador só recorre a um movimento aleatório quando nenhuma das condições for satisfeita.

Figura 10: Fluxograma de uma jogada ofensiva e defensiva



Fonte: Elaborada pelo autor

Com a finalidade de averiguar o comportamento de um jogador usando esta abordagem em uma disputa contra um agente humano, todas as jogadas estratégicas foram utilizadas de forma bastante eficiente, sendo que as jogadas ofensivas de tentativa de bloqueio se mostraram frequentemente como as mais evidentes.

5 RESULTADOS E DISCUSSÕES

Foram realizadas excessivas simulações de partidas do jogo com diferentes agentes competindo entre si, com intuito de analisar seus comportamentos em diversas situações e verificar eficácia dos seus resultados ou o grau de sucesso com que atingiram seus objetivos. Para aumentar a precisão dos dados resultantes, foram feitas 10.000 simulações para cada disputa de oponentes ao longo de dez etapas. Em cada etapa, foi obtido um percentual de vitórias, derrotas e empates. No final, é calculado a média geral de todos os percentuais obtidos ao longo das etapas.

As primeiras simulações consistem da disputa entre dois agentes randômicos. Pode-se perceber pela média geral da Tabela 1 que ambos apresentam comportamento levemente similar, tendo apenas as pedras distribuídas de forma aleatória como elemento diferencial uma vez que nenhum deles utiliza qualquer estratégia. Os resultados das vitórias, derrotas e empates estão sob a perspectiva de primeiro jogador.

Tabela 1: Agente randômico vs. Agente randômico

Simulação	Vitórias (%)	Derrotas (%)	Empates (%)
0-1000	49.4	49.9	0.7
1000-2000	51.2	48.4	0.4
2000-3000	49.8	48.9	1.3
3000-4000	48.3	50.7	1.0
4000-5000	49.9	49.6	0.5
5000-6000	50.5	48.8	0.7
6000-7000	46.0	52.7	1.3
7000-8000	49.2	50.3	0.5
8000-9000	49.0	49.7	1.3
9000-10000	52.6	46.6	0.8
Média	49.6	49.6	0.8

Fonte: Dados organizados pelo autor

Para as simulações de agente defensivo contra um agente randômico, os resultados mostraram um ligeiro aumento no número de vitórias com o uso de heurística. Estratégias de defesa aplicadas ao agente permitiram que ele efetuasse movimentos defensivos ao invés de aleatórios. Vale a pena notar também que o número de empates se mostrou menos instável ao longo das etapas evidenciado pela mudança de comportamento de um dos agentes envolvidos. Esses resultados podem ser conferidos na Tabela 2. As vitórias, derrotas e empates estão sob a perspectiva de primeiro jogador, ou seja, o defensivo.

Tabela 2: Agente defensivo vs. Agente randômico

Simulação	Vitórias (%)	Derrotas (%)	Empates (%)
0-1000	58.7	40.6	0.7
1000-2000	59.4	40.0	0.6
2000-3000	62.0	37.7	0.3
3000-4000	59.2	40.0	0.8
4000-5000	57.7	41.6	0.7
5000-6000	61.5	37.8	0.7
6000-7000	62.0	37.3	0.7
7000-8000	60.4	38.9	0.7
8000-9000	59.0	40.2	0.8
9000-10000	62.5	37.0	0.5
Média	60.2	39.1	0.7

Fonte: Dados organizados pelo autor

A disputa de um agente ofensivo contra um aleatório mostrou resultados mais significantes de que o agente defensivo. Foi demonstrado através dos resultados da Tabela 3 que uma abordagem ofensiva, com o uso da heurística, apresenta melhores resultados contra um agente com movimentos aleatórios. As vitórias, derrotas e empates estão sob a perspectiva do agente ofensivo.

Tabela 3: Agente ofensivo vs. Agente randômico

Simulação	Vitórias (%)	Derrotas (%)	Empates (%)
0-1000	65.0	34.3	0.7
1000-2000	65.2	33.9	0.9
2000-3000	62.7	36.1	1.2
3000-4000	64.4	34.5	1.1
4000-5000	65.4	34.0	0.6
5000-6000	63.7	35.7	0.6
6000-7000	66.4	32.2	1.4
7000-8000	63.8	35.2	1.0
8000-9000	66.1	32.9	1.0
9000-10000	65.3	34.0	0.7
Média	64.8	34.3	0.9

Fonte: Dados organizados pelo autor

Com o intuito de averiguar o potencial da heurística combinando as abordagens de defesa e ofensividade contra um agente de movimentos aleatórios, pode-se concluir através da Tabela 4 que esta se torna a melhor das opções. As vitórias, derrotas e empates estão sob a perspectiva de primeiro jogador.

Tabela 4: Ofensivo e Defensivo vs. Randômico

Simulação	Vitórias (%)	Derrotas (%)	Empates (%)
0-1000	69.8	28.8	1.4
1000-2000	69.6	29.7	0.7
2000-3000	70.1	29.1	0.8
3000-4000	71.5	27.8	0.7
4000-5000	69.9	29.2	0.9
5000-6000	72.8	26.7	0.5
6000-7000	68.4	30.8	0.8
7000-8000	66.9	32.1	1.0
8000-9000	72.7	26.6	0.7
9000-10000	72.3	26.5	1.2
Média	70.4	28.7	0.9

Fonte: Dados organizados pelo autor

Para comprovar qual das abordagens, ofensiva ou defensiva, apresenta um melhor desempenho em uma disputa entre os dois, foi realizado o mesmo procedimento. Pela Tabela 5, entende-se que estratégias ofensivas são mais efetivas neste cenário. Vale a pena considerar também que o número de empates nesta disputa foi maior, quando comparado com as demais nas disputas anteriores. As vitórias, derrotas e empates da tabela estão sob a perspectiva de primeiro jogador.

Tabela 5: Ofensivo vs. Defensivo

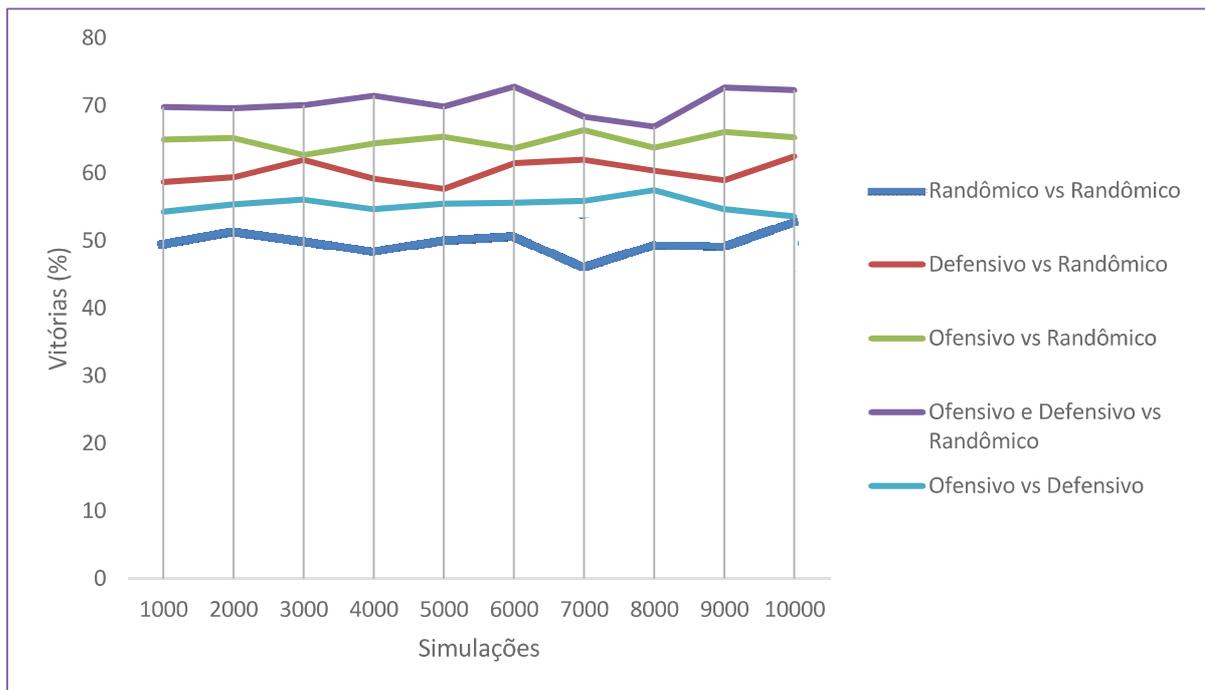
Simulação	Vitórias (%)	Derrotas (%)	Empates (%)
0-1000	54.3	44.7	1.0
1000-2000	55.4	43.5	1.1
2000-3000	56.1	42.6	1.3
3000-4000	54.7	44.5	0.8
4000-5000	55.5	43.6	0.9
5000-6000	55.6	43.4	1.0
6000-7000	55.9	43.3	0.8
7000-8000	57.5	41.7	0.8
8000-9000	54.7	44.3	1.0
9000-10000	53.6	45.6	0.8
Média	55.3	43.7	1.0

Fonte: Dados organizados pelo autor

Por fim, através do gráfico da Figura 11, evidencia-se um aumento considerável no número de vitórias do agente com o uso de heurística quando comparado com o agente randômico. A abordagem híbrida que combina estratégias ofensivas e defensivas confirma um aumento de 45,77% em relação ao número de vitórias obtido pelo agente randômico.

Considerando apenas a abordagem ofensiva em relação à defensiva, os dados concordam com os resultados obtidos por Reisman (1972) em seu trabalho sobre simulação de jogos, o qual confirma que estratégias ou métodos agressivos apresentam resultados significativamente melhores do que defensivos.

Figura 11: Percentual de vitórias das disputas sob perspectiva do primeiro agente



Fonte: Elaborada pelo autor

Conclui-se, desta forma, que o uso de uma abordagem heurística no jogo de dominós se mostra uma alternativa viável e bastante satisfatória, principalmente quando se pretende estudar o comportamento dos agentes através de suas interações no jogo bem como melhorar seus desempenhos. Comprovou-se também que o resultado da combinação de diferentes estratégias pode ser mais eficaz quando comparado com abordagens estratégicas isoladas.

6 CONSIDERAÇÕES FINAIS

O tema envolvido neste trabalho foi a proposição de um algoritmo heurístico com o intuito de aumentar a performance de um jogador ou um agente no jogo de dominós. Neste sentido, os resultados foram bastante significativos, tendo em vista que este jogo depende muito das propriedades de aleatoriedade e de informação incompleta. Mesmo que o agente seja dotado de diversas estratégias e uma base de jogadas clássicas do jogo, o seu desempenho ainda

dependerá das pedras que recebe que são distribuídas aleatoriamente. Uma solução ótima ou um estado de vitórias perfeito em um jogo como este se torna praticamente impossível, todavia, esta peculiaridade faz deste jogo um ambiente favorável para aplicação de métodos heurísticos.

Para trabalhos futuros, sugere-se a aplicação de Aprendizado de Máquina (Machine Learning) com o intuito de dotar o agente de jogo com uma habilidade de aprender outros movimentos e modificar suas próprias estruturas de decisão internas, adequando-se assim a novos cenários. Além disso, o desenvolvimento de uma interface gráfica favoreceria uma melhor interação com diversos agentes humanos. Para tal, serão necessários estudos mais aprofundados e uma flexibilidade maior de tempo para implementação.

ARTIFICIAL INTELLIGENCE IN BOARD GAMES: PROPOSITION OF HEURISTIC FOR THE GAME OF DOMINOES

ABSTRACT

This work aims to analyze, develop and apply Artificial Intelligence techniques to one of the most popular board games in history: Dominoes. Thus, a heuristic algorithm is proposed in order to increase the performance of a machine player or an agent in the game environment. To validate the algorithm, tests were performed which consisted of simulations of matches between several agents, including a human agent. Data resulting from excessive interaction or competition between agents were quite satisfactory, presenting an increase of 45.77% in the number of victories by using the heuristic algorithm. The data are analyzed and discussed in this paper.

Keywords: Heuristic Algorithm. Dominoes. Increase of Performance.

7 REFERÊNCIAS

BUENO, F. Métodos Heurísticos: Teoria e implementações. IFSC, Araranguá, 2009.

CELKO, J. Pagat: Card, games and rules, 2001. Disponível em: <http://www.pagat.com/tile/wdom/history.html>.

DINIS, F. Artificial Intelligence Techniques in Games with Incomplete Information: Opponent Modelling in Texas Hold'em Master's Thesis submitted to Faculdade de Engenharia da Universidade do Porto, 2008. Disponível em: <http://paginas.fe.up.pt/~ee07056/Artificial%20Intelligence%20Techniques.pdf>

IBM RESEARCH. Intelligent Agents Project at IBM T. J. Watson Research. Disponível em: site IBM Corporation, 1998.

JEFFREY, S. R. Baseado em An Introduction to multiagent systems por Michael Wooldridge, 2002. Disponível em:
<http://www.cs.ox.ac.uk/people/michael.wooldridge/pubs/imas/distrib/pdf-slides/lect02.pdf>.
Acesso em: 3 de Fevereiro de 2016.

MAES, P. Intelligent Software: Easing the Burdens that Computers put on people. IEEE Expert 11(6): 62-63, 1996.

MCLEOD, J. Pagat: Card, games and rules, 2008. Disponível em:
<http://www.pagat.com/tile/wdom>. Acesso em: 10 de Fevereiro de 2016.

REISMAN, S. Simulated Strategies of Game Playing, IBM Canada, Ltd. Laboratory, 1972.

RUSSELL, J., and P. NORVIG. **Artificial Intelligence: A Modern Approach**. Upper Saddle River, NJ: Prentice Hall, 1995.

SHANNON. C. E. Programming a Computer for Playing Chess. Philosophical Magazine, Ser.7, Vol. 41, No. 314 - March 1950.

SILVA, D. De onde vem as palavras: origens e curiosidades da língua portuguesa. pag. Ed. Editora Lexicon, 2014 Philosophical Magazine, Ser.7, Vol. 41, No. 314 - March 1950. Disponível em: <https://books.google.com.br/books?id=P3JxBAAAQBAJ>. Acesso em: 10 de Fevereiro de 2016.

SIMON, Herbert A. Comportamento Administrativo: estudo dos processos decisórios nas organizações administrativas. Rio de Janeiro: FGV, 1970.

SIMON, Herbert A.; NEWELL, A. Heuristic Problem Solving: the next advance for operations research. Operations Research, n. 7, p. 1-10, 1958.

STORMDARK I.P. and Media: Domino history. Disponível em: <http://www.domino-play.com/Strategy>, 2015. Acesso em: 10 de Fevereiro de 2016

WALKLATE R. J., Applying Suitable Artificial Intelligence Techniques to a Game of Dominos, Bachelor of Science in Computer Science with Honors The University of Bath, 2007.

WINSTON, P. H. Artificial Intelligence. Addison Wesley (Third edition), 1992.

WOOLDRIDGE, Michael; JENNINGS, Nicholas R. Towards a Theory of Cooperative Problem Solving. Sixth European Workshop on Modelling Autonomous Agents in Multi-Agents Worlds, 1995.