

# ÁREAS DE PESQUISA E TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL EM JOGOS DIGITAIS

Mariana de Luca Reis<sup>1</sup>  
Kleber de Oliveira Andrade<sup>2</sup>

doi: 10.47283/244670492022100171

## Resumo

A Inteligência Artificial (IA) é um recurso bastante utilizado nos Jogos Digitais. O objetivo deste artigo é apresentar uma análise sobre as diferentes abordagens que podem ser utilizadas para se implementar Inteligência Artificial em Jogos. Para isso, é feita uma revisão da literatura sobre o tema, destacando brevemente o histórico e evolução da IA, como a IA e os jogos se relacionam, elencando as principais áreas de pesquisa sobre este tópico, quais suas abordagens e diferentes técnicas de aplicação. Além disso, o artigo traz discussões a respeito da IA que existe atualmente poder ou não ser considerada Inteligência e como o desenvolvimento das técnicas de IA em jogos podem contribuir com o desenvolvimento da Inteligência Artificial Geral.

**Palavras-chaves:** Inteligência Artificial. Jogos Digitais. Técnicas de Inteligência Artificial.

## Abstract

*The Artificial Intelligence (AI) is a resource widely used in Digital Games. The goal of this article is to present an analysis about different approaches that can be used to implement Artificial Intelligence in Games. To accomplish that, a literature review about the subject is made, briefly highlighting the history and evolution of AI, how AI and games are related, listing the main research areas on this topic and which are the different approaches and application techniques. Furthermore, the article brings discussions about whether the AI that currently exists may or may not be considered actual Intelligence and how the development of AI techniques used in games can contribute with the development of General Artificial Intelligence.*

**Keywords:** Artificial Intelligence. Digital Games. Artificial Intelligence Techniques.

## Introdução

A Inteligência Artificial (IA) é um recurso amplamente utilizado quando falamos em Jogos Digitais. Dentro de um jogo, a IA pode desempenhar diferentes papéis, como controlar personagens não jogáveis, gerar cenários e fases, simular o comportamento de um adversário e mesmo guiar a história do jogo.

De uma maneira ou de outra, Inteligência Artificial sempre esteve relacionada a Jogos, sejam eles digitais ou não. Por exemplo, de acordo com (FRANCHI, 2005), existe uma analogia de que o jogo de xadrez é a *Drosófila* da Inteligência Artificial – assim como a mosca da fruta e seu rápido ciclo reprodutivo acaba sendo um dos principais objetos de estudo para testar

<sup>1</sup> Tecnóloga em Jogos Digitais – Fatec Americana. E-mail: marianadeluca.reis@gmail.com

<sup>2</sup> Docente – Fatec Americana. E-mail: pdjkleber@gmail.com

teorias genéticas, o jogo de xadrez com suas regras bem definidas e complexidades foi muitas vezes utilizado para testar teorias de Inteligência Artificial. Alan Turing, considerado o principal “pai” da Ciência da Computação, antes mesmo de que qualquer computador pudesse ser inventado, criou um algoritmo de *Minimax* e o testou para jogar xadrez, utilizando apenas caneta e papel (TOGELIUS, 2016). O xadrez foi usado tantas vezes como referência para implementar Inteligência Artificial ao longo dos anos que culminou com o computador *Deep Blue*, desenvolvido pela IBM, ganhando do enxadrista campeão mundial, Garry Kasparov, em 1997. E o motivo para encontrarmos tantas referências especificamente ao jogo de Xadrez é que, para ser capaz de jogá-lo, supõe-se que o jogador precise de uma certa “inteligência”. Mas, afinal, o que é Inteligência? Ou, ainda, o que podemos chamar de Inteligência Artificial?

Através de pesquisas relacionadas ao tema, tendo como fontes livros, artigos publicados e sites, procura-se trazer algumas definições a respeito dessas questões, além de apresentar quais as principais áreas de pesquisa sobre Inteligência Artificial voltada para jogos e quais as técnicas de IA mais prevalentes que podem ser encontradas. Como essa indústria está em expansão, há o surgimento de jogos mais sofisticados que demandam uma Inteligência Artificial mais complexa para se tornarem atrativos, o que, por sua vez, requer que novas técnicas sejam desenvolvidas nessa área. E esse é outro ponto de discussão: o que o avanço da tecnologia de IA em jogos pode representar para pesquisas de Inteligência Artificial Geral.

Este artigo se divide em 4 seções: inicialmente relata-se alguns fundamentos sobre Inteligência Artificial e um pouco de seu histórico; em seguida, na seção 2, a relação entre IA e os jogos é evidenciada, com destaque para as principais áreas de pesquisa sobre o tema. A seção 3 do artigo identifica as técnicas ou métodos de IA predominantes nos jogos digitais. Por fim, a seção 4 busca elucidar as vantagens de se usar os jogos eletrônicos como plataformas de teste para IA, apontando como isso pode contribuir para a evolução da Inteligência Artificial Geral.

## **1 Inteligência artificial**

A Inteligência Artificial (IA) é a habilidade de realizar tarefas geralmente associadas a seres inteligentes, mas desempenhadas por um computador (COPELAND, 2020); ou, ainda, segundo artigo da *IBM Cloud Education*, a Inteligência Artificial é a utilização de computadores e máquinas para imitar a capacidade de resolução de problemas e tomada de decisões da mente humana.

Enquanto a ideia de uma “máquina que pensa” remonta até a Grécia Antiga, é a partir do advento da computação e com o artigo publicado em 1950 por Allan Turing, “*Computing Machinery and Intelligence*”, que consideramos o nascimento da IA. É neste artigo que ele propõe o seguinte questionamento: “As máquinas podem pensar?”; e onde o famoso “Teste de Turing” é apresentado: um interrogador humano tentaria distinguir se as respostas às suas perguntas foram elaboradas por outro humano ou por um computador. Contudo, o termo “Inteligência Artificial” só foi cunhado em 1956 por John McCarthy, na primeira conferência sobre IA na Universidade de Dartmouth. Segundo ele, Inteligência Artificial é “a ciência e engenharia capaz de construir máquinas inteligentes”.

Já sobre o que é inteligência, enquanto mesmo a mais simples das ações observadas em humanos é vista como resultado de uma mente inteligente, o mais complicado dos comportamentos notados em um inseto nunca é tido como um indicativo de inteligência, mas de instinto; a principal diferença é que, de acordo com a psicologia, o que caracteriza inteligência não é apenas um traço específico, mas um conjunto de muitas e diversas habilidades combinadas (COPELAND, 2020). De qualquer maneira, segundo Ribeiro *et al* (2013) esse é um tema controverso: “se inteligência artificial é a inteligência aplicada às máquinas, o que é exatamente “inteligência”, independentemente se ela está ou não associada a seres vivos?”.

Por exemplo, supõe-se que todo ser humano possua inteligência. Entretanto, tomando o exemplo do jogo de Xadrez, são considerados bons jogadores aqueles humanos com uma inteligência acima da média. Kasparov, o campeão mundial que perdeu para a Inteligência Artificial, levantou a seguinte questão: se o computador fez o mesmo movimento que um humano faria por razões completamente diferentes, ele fez um movimento inteligente? A inteligência de uma ação depende de quem (ou do que) a faz? (DOS ANJOS, 2013).

Devido a essas indagações, questões envolvendo Inteligência e Inteligência Artificial também fazem parte do ramo da filosofia, cuja abordagem não é o objetivo deste artigo. Porém, nesse sentido filosófico, Franchi (2005, p. 3) define Inteligência Artificial como uma teoria que identifica algumas das mais básicas características da subjetividade humana com os procedimentos básicos necessários para jogar um jogo como Xadrez. De acordo com Copeland (2020), as pesquisas sobre Inteligência Artificial focam nos seguintes componentes da inteligência humana: aprendizagem, raciocínio, resolução de problemas, percepção e uso da linguagem. Ao longo do tempo, muitas pesquisas e experimentos foram desenvolvidos. Por exemplo, segundo o filósofo norte americano John Searle, podemos classificar a Inteligência Artificial em dois tipos: IA forte e IA fraca.

Diz-se que uma Inteligência Artificial é do tipo “fraca” ou “restrita” é aquela que foi treinada para e cujo foco é realizar tarefas específicas. Segundo Searle, mesmo que um computador execute com sucesso um algoritmo, isso não significa que a máquina tenha “entendido” o que estava fazendo; há apenas a “impressão de inteligência”, ela não é consciente e representa somente a execução de processos humanos (DOS ANJOS, 2013). É esse tipo de inteligência que encontramos muitas vezes em jogos, é o tipo de IA utilizada pela *Siri (Apple)*, *Alexa (Amazon)*, pelo *IBM Watson* e por carros autônomos.

Já a Inteligência Artificial “forte” é uma forma mais teórica de IA e se refere a máquinas que possuem uma inteligência equivalente à humana, que efetivamente pensam como humanos: máquinas autoconscientes, com capacidade de raciocínio intuitivo e com a habilidade de resolver problemas, aprender e planejar o futuro. Em resumo, seu objetivo é o desenvolvimento de máquinas cuja habilidade intelectual seja indistinguível à de um ser humano (COPELAND, 2020). Os exemplos desse tipo de Inteligência Artificial atualmente só podem ser encontrados na ficção, como o robô HAL de *2001: Uma Odisseia no Espaço* (1968) ou a assistente virtual Samantha do filme *Ela* (2013).

De acordo com Funge e Milington (2009), quando estudamos sobre Inteligência Artificial sob uma perspectiva acadêmica geral, podemos dividi-la em três períodos:

a) Primórdios da IA: inclui um tempo anterior ao surgimento dos computadores e trazia indagações filosóficas, como “O que produz os pensamentos?”, “É possível dar a vida a um objeto inanimado?” e “Qual a diferença entre o cadáver e o humano que ele era anteriormente?”. Foram os esforços investidos para “quebrar” códigos inimigos durante a 2ª Guerra Mundial, em 1940, e realizar cálculos necessários para competir na iminência de uma guerra atômica que motivou o desenvolvimento dos primeiros computadores programáveis – com os pioneiros das ciências da computação sendo também os pioneiros da IA, como von Neumann, Shannon e principalmente, Turing;

b) Era Simbólica: do final da década de 1950 até o início dos anos 1980, o principal impulso das pesquisas em IA eram sistemas “simbólicos” – aquele em que o algoritmo é dividido em dois componentes: um conjunto de conhecimentos (representado por símbolos como palavras, números, frases ou imagens) e um algoritmo de raciocínio que manipula esses símbolos para criar combinações de símbolos, que se espera que representem soluções para os problemas ou novos conhecimentos. Uma abordagem simbólica aplicável aos jogos inclui *path finding*, árvores de decisão, máquinas de estado e algoritmos de direção;

c) Era Moderna: gradualmente ao longo da década de 1980 até o início de 1990, como o sucesso da IA simbólica em resolver problemas simples não escalou de maneira satisfatória para a resolução de problemas mais difíceis, ou para lidar com situações de incerteza e complexidade do mundo real, houve uma frustração crescente com esse tipo de abordagem. O resultado foi um movimento em direção à computação natural: técnicas inspiradas em biologia ou outros sistemas naturais, como redes neurais, algoritmos genéticos e a meta-heurística para otimização conhecida como recozimento simulado. Aos poucos os pesquisadores perceberam que o ingrediente chave dessa nova abordagem não era tanto a conexão com o mundo natural, mas sim a habilidade de lidar com incertezas e a importância que isso representa na resolução de problemas da realidade.

Mesmo com a evolução da Inteligência Artificial na era moderna, isso não significou o fim das abordagens simbólicas. Por exemplo, aplicativos de engenharia da computação estatística sempre usam tecnologia simbólica. Quanto mais restrito é o domínio do problema que deve ser resolvido, mais fácil será desenvolver um algoritmo eficiente, o que significa que muitas vezes não é necessária uma abordagem tão sofisticada. Algoritmos de raciocínio utilizados na IA simbólica geralmente consistem em buscas: tentativas de diferentes possibilidades para obter o melhor resultado. E isso nos traz a regra de ouro da IA: busca e conhecimento estão intrinsecamente ligados – quanto mais conhecimento se tem, menos buscas será preciso realizar para chegar a uma resposta; quanto maior a rapidez ou capacidade de se buscar resultados, menos conhecimento será necessário para realizá-lo – busca e conhecimento podem ser entendidos como dois lados da mesma moeda. E, embora isso esteja mudando, a grande maioria da IA utilizada em jogos ainda usa tecnologia simbólica (FUNGE e MILINGTON, 2009).

## **2 Inteligência artificial nos jogos digitais**

Inteligência Artificial (IA) de uma forma ou outra, pode ser encontrada em qualquer jogo eletrônico, seja controlando personagens não jogáveis (NPC's) ou nos demais aspectos do

mundo do jogo. Segundo Lowood (2019, p. 1, tradução nossa), “A ideia de jogar jogos em computadores é quase tão antiga quanto o próprio computador”.

Data de 1950 a primeira proposta de ensinar computadores a jogar xadrez, feita pelo matemático e engenheiro Claude Shannon e, como já mencionado no artigo, essa proposta estimulou décadas de pesquisa em xadrez e programas capazes de jogá-lo (LOWOOD, 2019). Já os primeiros jogos eletrônicos surgiram de laboratórios de universidades e indústrias de informática, sendo um exemplo disso o jogo *Spacewar!* de 1962, criado por Steve Russel no MIT (*Massachusetts Institute of Technology*) (LOWOOD, 2019). Outro exemplo de um dos primeiros jogos para plataformas digitais mais famosos é o jogo *Pong*, da *Atari* (1972). Esses jogos, entretanto, só poderiam ser jogados por dois jogadores disputando um contra o outro (*versus mode*).

Com o aumento da sua comercialização, a indústria percebeu que seria interessante permitir também que esses jogos pudessem ser jogados por apenas uma pessoa contra o computador. E o grande desafio que se apresentou foi como implementar esses adversários de forma que agissem de maneira inteligente, mantendo o nível de desafio e diversão para o jogador que resolvesse jogar sozinho (DOS ANJOS, 2013).

As primeiras técnicas de IA desenvolvidas não possuíam nada de realmente “inteligente” em seus comportamentos, sendo conhecidas como “programação de jogabilidade” (SCHWAB, 2004 *apud* KISHIMOTO, 2004, p. 4). Com a evolução dos jogos digitais, o desenvolvimento de novas técnicas de Inteligência Artificial tornou-se necessário. A tabela a seguir apresenta diferentes técnicas de IA utilizadas em jogos ao longo do tempo:

**Tabela 01: Evolução da IA utilizada em Jogos**

Ano	Jogo: Descrição e Aplicação da IA	IA utilizada
1962	<i>Spacewar!</i> : considerado primeiro jogo de computador, requeria 2 jogadores e foi escrito para o microcomputador PDP-1.	-
1972	<i>Pong</i> : versão eletrônica do tênis de mesa que requeria 2 jogadores.	-
1974	<i>Pursuit, Qwak!</i> : jogos simulação de vôos e tiro ao alvo com padrões de movimentação em que os jogadores tinham que atirar em alvos móveis.	Padrões de movimento.
1975	<i>Gun Fight</i> : jogo de duelo de tiros, para dois jogadores, com movimentação aleatória.	Padrões de movimento.
1978	<i>Space Invaders</i> : jogo de tiro com inimigos com movimentos padronizados, que atiram contra o jogador.	Padrões de movimento.
1980	<i>Pac Man</i> : conta com padrões de movimentos dos fantasmas inimigos, porém cada um deles possui uma “personalidade” diferente sobre o modo em que caça o jogador.	Padrões de movimento.



1984	<i>Karate Champ</i> : um dos primeiros jogos de luta <i>singleplayer</i> contra o computador.	Padrões de movimento.
1990	<i>Herzog Zwei</i> : primeiro jogo de estratégia em tempo real. É a primeira vez em um algoritmo de busca de caminho é implementado, mostrando-se previsível com algum tempo de jogo.	Máquinas de estados finitos (ou <i>Finite State Machines</i> , FSM's em inglês).
1992	<i>Doom</i> : primeiro jogo de tiro em primeira pessoa.	Máquina de estados.
1996	<i>BattleCruiser: 3000AD</i> : primeiro uso de redes neurais em um jogo comercial.	Redes neurais.
1997	<i>Deep Blue</i> derrota o atual campeão de xadrez Gary Kasparov.	Várias técnicas.
1998	<i>Half-Life</i> : até então a melhor inteligência artificial em jogos, uso de linguagens de script.	Máquina de estados / <i>Script</i> .
2001	<i>Black &amp; White</i> : utiliza criaturas que usam aprendizado por reforço e observação ( <i>observational learning</i> ) e redes neurais para aprender com as decisões tomadas pelo jogador.	Várias técnicas.
2014	<i>Alien Isolation</i> : algoritmos de árvore de decisão que vai sendo desbloqueada conforme o jogo avança. O <i>Alien</i> adapta-se à maneira de jogar do jogador: o personagem apresenta comportamentos mais complexos “aprende” determinados traços do estilo do jogador.	Algoritmos de busca em árvore, sistemas de busca pré-programados.

Fonte: Adaptado de Dos Anjos, 2013 e Schwab, 2004 *apud* Kishimoto, 2004, com informações adicionais.

É relevante salientar que a indústria de *games* não costuma divulgar abertamente quais técnicas de IA são utilizadas em seus jogos, nem a maneira como os algoritmos foram implementados. Isso não deixa de ser um dificultador para pesquisas sobre o tema.

### 2.1 Áreas de Pesquisa sobre Inteligência Artificial em Jogos Digitais

As pesquisas sobre IA em jogos visam resolver problemas e permitir inovações que englobam o *game design*, o desenvolvimento de jogos e a jogabilidade em si. O principal foco é criar mecanismos de jogo mais inteligentes e eficientes, fazendo com que os *games* se tornem mais imersivos, interessantes e divertidos. (LUCAS *et al.*, 2012).

A IA utilizada em pesquisas acadêmicas e aquela empregada em jogos possuem objetivos diferentes. Enquanto a primeira busca a solução para problemas complexos (principalmente quando se trata de uma IA do tipo “forte”), o principal objetivo da segunda é a diversão (KISHIMOTO, 2004). Entretanto, segundo Togelius (2016) as pesquisas de IA em jogos podem ser muito úteis para pesquisadores de IA acadêmica. Como para construir uma Inteligência Artificial completa é necessário que ela seja aplicada em um sistema que aja em algum tipo de ambiente, a IA acadêmica muitas vezes utiliza robôs – que são caros para serem

construídos, complexos e lentos. Já os jogos são mais fáceis e baratos de serem desenvolvidos, podem ser acelerados em várias vezes sua velocidade original e é mais rápido de se iniciar os testes com a IA, com os experimentos podendo ser repetidos por milhares de vezes sucessivamente. Em contrapartida, a evolução em pesquisas sobre IA acadêmica avançada também significaria uma grande evolução para os jogos, com a possibilidade de jogos inteiros sendo criados a partir de Inteligência Artificial.

A análise das áreas de pesquisa descritas neste artigo foca em sua aplicação nos jogos digitais, já que o objetivo é salientar a relação destes com a Inteligência Artificial. Durante os últimos 15 anos, com o estabelecimento da área Inteligência Computacional<sup>3</sup> em jogos como um campo de pesquisa a parte, surgiram diversos encontros anuais sobre o tema, como o *IEEE Conference on Computational Intelligence and Games (CIG)* e a *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. Durante um desses encontros, no seminário *Dagstuhl Seminar on Artificial and Computational Intelligence in Games*, pesquisadores foram convidados a identificar e discutir os futuros avanços das áreas-chave de IA em jogos (YANNAKAKIS e TOGELIUS, 2015). Conforme relatado pelos autores citados, as dez principais áreas da Inteligência Artificial e Inteligência Computacional identificadas estão listadas a seguir. É importante salientar que a popularidade de cada método de IA utilizado nessas diferentes áreas está intimamente ligado à tarefa a ser realizada ou o objetivo que se deseja alcançar.

1. Aprendizagem de comportamento por NPCs (*Non Playable Characters* ou Personagens não jogáveis)

O foco é o uso de técnicas de aprendizagem por reforço. Basicamente, aprender a jogar algum jogo pode ser encarado como um problema de aprendizagem por reforço. Algoritmos evolutivos é o método dominante para a aprendizagem de comportamento de NPCs, mas outras técnicas incluem: coevolução competitiva, recozimento simulado, aprendizagem por diferenças temporais, algoritmos de otimização e mesmo combinações entre esses métodos.

2. Busca e planejamento

Um dos algoritmos fundamentais da ciência da computação, o algoritmo de busca em jogos é também bastante utilizado em aplicações de IA para jogos de tabuleiro. Em jogos digitais, dois tipos de busca são particularmente importantes: busca informada (*best-first search*), como o algoritmo A\* (utilizado na técnica de *path finding*) e buscas em árvore como o algoritmo de busca *MiniMax*, dois exemplos bem antigos. O planejamento, por sua vez, é a aplicação da busca no intervalo entre estados, procurando o caminho mais curto de um estado para outro. É uma das áreas de pesquisa que mais se conectam e influenciam outras.

3. Modelagem de jogador

---

<sup>3</sup> A Inteligência Computacional é uma área da Computação Natural que busca desenvolver sistemas computacionais capazes de resolver problemas complexos, através da observação da capacidade da natureza humana em resolver problemas (VALLE, 2011 *apud* BARBOSA, VEIGA e CARVALHO, 2012). Pode ser vista como um ramo da Inteligência Artificial que estuda mecanismos adaptativos que simulam aspectos da inteligência humana, como a percepção, o raciocínio, potencial para aprendizagem etc., e incluem áreas de pesquisa como redes neurais, teoria dos conjuntos nebuloso e computação evolucionária.

Nesta área de pesquisa, modelos computacionais são criados para detectar como um jogador percebe e reage ao *gameplay*. Geralmente esses modelos são criados usando métodos de *machine learning*, onde os dados que se relacionam com algum aspecto da interação jogador-jogo associam-se a rótulos derivados de alguma avaliação da experiência ou do efeito produzido no jogador; esses dados são coletados a partir de medições fisiológicas ou questionários. É uma área também preocupada em observar estruturas comportamentais do jogador, mesmo sem correlações com experiência disponíveis (como ao identificar tipos de jogador). Um exemplo clássico que utiliza essa área como parte do *gameplay* é o modo como são criadas as expressões faciais dos avatares na série *The Sims (MAXIS, 2000)*.

#### 4. Jogos como modelos para IA

São jogos que oferecem interfaces para sistemas de IA externos, de maneira que se possa avaliar a performance desses sistemas nas tarefas associadas à IA utilizada como modelo. Muitas vezes esse tipo de pesquisa está ligado a competições realizadas em conferências sobre esses modelos. Ao mesmo tempo que ter bons modelos é propício para se fazer comparações entre os métodos utilizados, aperfeiçoando-os, pode acabar concentrando muitos esforços na resolução de problemas demasiado específicos, em detrimento das pesquisas como um todo (por exemplo, há poucos problemas desafiadores e modernos na área de aprendizagem por reforço, o que dificulta a comparação dos algoritmos criados). De qualquer maneira, os jogos são ideais para esses modelos se basearem, por possuírem vantagens como funções de pontuação diversas, poderem ser instalados como aplicativos *standalone* e incluírem desafios adequados para humanos, o que também os faz desafiadores para os métodos de IA.

#### 5. Geração procedural de conteúdos (PCG – Procedural Content Generation)

PCGs se referem a criação de conteúdo para jogos de maneira automática ou semiautomática, como *levels*, fases, mapas, itens, missões e texturas. Recentemente essa área viu uma expansão em como criar PCGs de maneira mais controlada para diversos tipos de conteúdo para jogos, mas desde a década de 1980 esse recurso pode ser encontrado, mesmo que em um papel limitado. As técnicas mais utilizadas para a criação de PCGs são buscas evolucionárias e solução de restrições. Também está relacionada com as áreas de criatividade computacional e design de interação, entre outras.

#### 6. Narrativa computacional

As narrativas dos jogos podem desempenhar um papel essencial na criação de sua estética, que por sua vez pode ter impacto nos aspectos cognitivos e afetivos da experiência proporcionada ao se jogar. O estudo dessa área irá focar nos aspectos representacionais e geracionais das histórias, já que elas podem ser contadas através de jogos. Pode se relacionar bastante com a área de PCG, já que a representação da história narrada e a história em si são vitais para a experiência do jogador, e a geração automática ou semiautomática dessas narrativas pertencem a área de *storytelling* interativa, que é considerada uma forma de PCG baseado em histórias.

#### 7. Agentes Verossímeis

Envolve o estudo de verossimilhança em jogos e as investigações dos mecanismos para a construção de arquiteturas que pareçam ter características verossímeis ou próprias de humanos,



orientadas a um objetivo. As abordagens para a construção dessas arquiteturas podem ser tanto *bottom-up* quanto *top-down*, tentando imitar a maneira como humanos jogam. Esse tipo de pesquisa é central para os jogos que incorporam alguma forma de NPCs.

#### 8. Game design auxiliado por IA

Segundo os autores, é uma das áreas de pesquisa mais promissoras para o desenvolvimento de jogos melhores. Essa área se concentra na criação de ferramentas impulsionadas por inteligência artificial que suportem o processo de desenvolvimento e design de jogos, podendo ajudar na criação de conteúdos que variam de *levels* e mapas a mecânicas de jogo e narrativas.

#### 9. Inteligência Artificial geral de jogo

Diz respeito ao desenvolvimento de agentes capazes de jogar uma grande variedade de jogos após algum processo de aprendizagem, e não apenas um tipo de jogo específico (objetivo para o qual geralmente as IAs são criadas - muitas com técnicas tão específicas que não conseguem desempenho satisfatório nem mesmo em jogos parecidos ou em variações de um mesmo jogo). Pesquisadores de inteligência artificial definiram essa área de pesquisa devido à fragmentação da IA moderna e com a falta de interesse em suas “grandes questões”. Existem competições anuais sobre esse tema, em que competidores tentam desenvolver agentes desse tipo para serem testados em jogos genéricos diversos; nos últimos anos, os vencedores utilizaram algoritmos baseados nas Árvores de Busca de Monte Carlo.

#### 10. IA em jogos comerciais

Os métodos de IA utilizados em jogos comerciais e aqueles aplicados em pesquisas de IA em jogos divergem em objetivos e resultados, já que estão focados em diferentes problemas: enquanto comercialmente o que se quer desenvolver é algo que funcione suficientemente bem e que pareça bom para o jogador dentro de um determinado contexto, os pesquisadores acadêmicos buscam soluções gerais para problemas mais profundos e genéricos. Entretanto, conforme já foi destacado neste texto, ambos podem se beneficiar um do outro; métodos comerciais podem servir de inspiração para pesquisas acadêmicas e os resultados que elas proporcionam podem ser aplicados em jogos comerciais.

Essas áreas se relacionam e influenciam entre si. Várias delas citam algumas das técnicas de IA que serão descritas na seção seguinte do artigo.

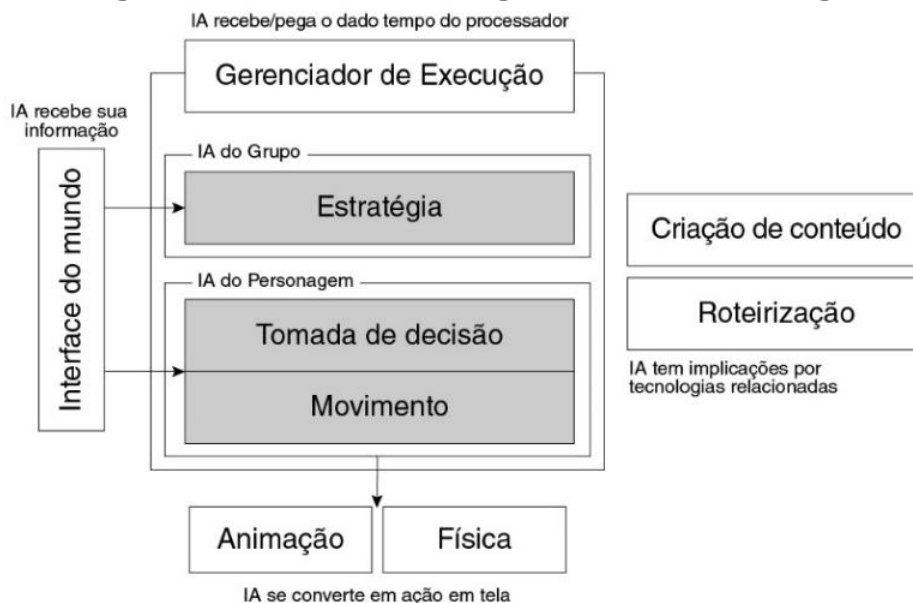
### 3 Técnicas de inteligência artificial aplicada em jogos

Para saber qual técnica ou método de IA aplicar em um jogo, o desenvolvedor não precisa ter um conhecimento aprofundado sobre todas elas, mas é interessante ter uma base sobre cada uma para saber aplicar a estratégia de IA mais adequada a seus jogos (FLAUSINO, 2008).

Antes, porém, para facilitar o entendimento da IA usada em jogos, Milington e Funge (2009, p. 8) propõem um modelo de estrutura consistente no qual, em geral, todas as técnicas se encaixem. Este modelo separa as tarefas da IA (ou personagens/elementos controlados por ela) em três seções: movimento, tomada de decisão e estratégia. As duas primeiras englobam algoritmos que estão relacionados aos personagens individualmente (IA do personagem na figura abaixo), já a terceira opera em todo um time ou lado (IA do grupo). Ao redor desses

componentes, há toda uma infraestrutura adicional. Nem todos os jogos requerem essas três seções: ao jogar xadrez, a IA não se importa com tomada de decisões ou se preocupam sobre como se mover, suas decisões vêm a partir do nível de estratégia. Por outro lado, muitos jogos não precisam da camada estratégica, sendo apenas reativos, com a IA tomando pequenas decisões e agindo conforme a decisão tomada.

**Figura 01 – Modelo de Inteligência Artificial em Jogos**



Fonte: Adaptado de Milington e Funge (2009, p. 9)

A camada de movimento se refere a algoritmos que transformam decisões em algum tipo de deslocamento. Por exemplo, a decisão de um inimigo se mover até o jogador para atacá-lo; ou então personagens desviando de obstáculos. Ainda segundo os autores, muitas ações acontecem usando animações diretamente, como no jogo *The Sims*: quando o personagem está sentado na mesa diante de um prato de comida pretendendo realizar a ação de comer, o jogo simplesmente mostrará a animação correspondente, com o IA se limitando a decidir que a animação deverá ser exibida. Assim que isso ocorre, não há mais necessidade da IA. Entretanto, se o personagem estiver em outro local e decidir comer, é a IA que decide que ele deverá se dirigir até a fonte de comida.

Já a tomada de decisão diz respeito ao personagem decidindo o que fazer em seguida, como atacar, permanecer parado, se esconder, explorar, evadir, patrulhar etc. de acordo com o que for mais apropriado para aquele momento do jogo; com o comportamento escolhido sendo executado usando a IA de movimento ou mostrando uma animação, ou seja, para que essa tomada de decisão ocorra é necessário algum tipo de deslocamento (um ataque de mão a mão, por exemplo, necessita que o personagem esteja próximo à vítima).

Por último, temos a camada de estratégia, que na abordagem dos autores se relaciona a comportamentos de grupos de personagens, com a IA controlando o comportamento de um conjunto deles.

Quaisquer que sejam os motivos que levem à necessidade de se implementar Inteligência Artificial em um jogo, serão as características do jogo que irão afetar na escolha e desenvolvimento do método de IA a ser efetivamente utilizado (YANNAKAKIS e TOGELIUS, 2018). As seguintes subseções têm como objetivo trazer uma visão geral das principais técnicas de Inteligência Artificial comumente encontradas nos jogos. Muitas vezes essas técnicas se misturam, com determinados algoritmos que representam algum tipo específico de método de IA sendo utilizado para implementação de outra técnica, por exemplo, uso de algoritmos genéticos para treinar uma rede neural.

### 3.1 Padrões de movimento

Os padrões de movimento, uma das primeiras técnicas de IA a serem utilizadas no surgimento jogos eletrônicos, nada mais são que diferentes manobras, movimentos predefinidos e comportamentos pré-programados ou pré-processados, na maioria das vezes utilizados para programar a animação de braços, pernas e algumas ações etc. dos personagens do jogo, dando a ilusão que esses personagens são inteligentes.

O algoritmo de padrões de movimento usa listas ou *arrays* de instruções codificadas, que irão dizer como o personagem deverá se mover ou se comportar a cada passo de um loop.

### 3.2 Perseguição e Evasão

Essa técnica também pode ser considerada como um padrão de movimento. Ao verificar a posição de um personagem *x* (predador) em relação a posição de um personagem *y*, o algoritmo de perseguição faz com que *x* avance em direção a *y* e o algoritmo de evasão, que *x* se distancie de *y* (LAMOTE, 1999 *apud* KISHIMOTO, 2004, p. 5).

Ou seja, o algoritmo deve primeiro descobrir a distância entre *x* e *y* e, a partir dela, incrementar essa distância ao perseguir e decrementá-la ao evadir.

De modo bastante simplista, podemos representá-lo:

**Figura 02 – Algoritmo de Perseguição e Evasão, representado em pseudocódigo**

Perseguição	Evasão
<pre> if (xPosicao &gt; yPosicao)     xPosicao--; else if (xPosicao &lt; yPosicao)     xPosicao++; </pre>	<pre> if (yPosicao &gt; xPosicao)     yPosicao++; else if (yPosicao &lt; xPosicao)     yPosicao-- ? &gt; ; </pre>

Fonte: Adaptado do artigo “*Game Creation with XNA AI Artificial Intelligence in Games*”, Wikibooks

Essa técnica pode usar a perseguição considerando o campo de visão ao invés da distância do personagem  $x$  em relação a  $y$ , fazendo com que a movimentação seja mais natural. Para isso são usados algoritmos que consideram o caminho mais direto e curto sem uso de passos desnecessários.

Outra abordagem seria utilizar um ponto de intercepção entre o personagem perseguidor e o que está sendo perseguido; para calcular esse ponto deve-se considerar três valores: posição, direção e velocidade.

Seguindo os passos:

1. Cálculo da velocidade relativa:  $v = vX + vY$
2. Distância relativa:  $d = posX - posY$
3. Tempo necessário para percorrer a distância a uma velocidade igual a velocidade relativa:  $t = \frac{|d|}{|v|}$
4. Posição prevista do personagem a ser interceptado:  $pos = posX + (vX \times t)$

**Figura 03 - Ponto de Intercepção**



Fonte: Adaptado do artigo "Game Creation with XNA AI Artificial Intelligence in Games", Wikibooks

### 3.3 Path finding (ou algoritmos de busca)

Essa técnica é uma das mais importantes aplicações da Inteligência Artificial em jogos comerciais (LUCAS *et al.*, 2012), consistindo em buscar por caminhos direcionando a movimentação de personagens. *Path finding* pode ser utilizado como:

- Desvio de obstáculos: simples de implementar, o desvio de obstáculos pode ser verificado na figura abaixo, em pseudocódigo.

#### Figura 04 - Representação da técnica de *Path finding* para desviar de obstáculos, em pseudocódigo

```
if (jogador na linha da visão)
{
    seguir o caminho direto em direção ao jogador
}
else
{
    (mover-se em direção aleatória)
}
```

Fonte: Adaptado do artigo “*Game Creation with XNA AI Artificial Intelligence in Games*”, Wikibooks

A abordagem acima pode ser bastante simples e não funcionar tão bem com a presença de muitos obstáculos; e uma solução seria a implementação de algoritmos que façam com que os personagens contornem obstáculos e se direcionem para um alvo assim que este entrar no seu campo de visão.

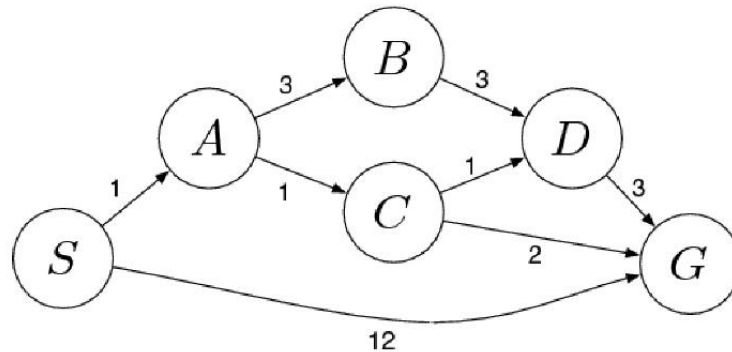
- *Path following*: não existe uma destinação definida para o personagem, mas sim uma rota predefinida em que ele deverá seguir. Exemplos dessa técnica podem ser encontrados em jogos de corrida de carros.
- *Breadcrumb path finding*: *breadcrumb* significa, em inglês, “migalhas de pão”. Nessa técnica de *path finding*, o próprio personagem do jogador define o caminho que será seguido pelo personagem NPC, como se, em cada passo, deixasse marcadores (migalhas de pão) pelo caminho, que serão seguidas.
- *Navegação Way Point*: (*way point*, “ponto de rota” em inglês) essa técnica posiciona nós pelo ambiente do jogo, e utiliza esses nós para decidir quais algoritmos de *path finding* implementar, de forma que eles se tornam mais “econômicos”. É útil em situações em que o ambiente é muito extenso e possui muitos obstáculos, já que os algoritmos de *path finding* podem acabar se tornando muito complexos.
- *A\**: um dos algoritmos mais famosos utilizados para implementar *path finding*, o algoritmo *A\**, combina técnicas de algoritmos de busca, como Busca Gulosa e Busca de Custo Uniforme. Trabalha classificando nós segundo um valor de  $f(n)$  e expandindo o nó com o menor custo. É representado como:

$$f(n) = g(n) + h(n)$$

Em que  $f(n)$  representa o custo total estimado do caminho iniciado em  $S$ , ou de uma transição, passar por  $n$  e atingir o nó final, sendo definido por  $g(n)$ , que representa a distância entre  $n$  e o nó inicial  $S$  (ou custo da transição em si), acrescido de  $h(n)$ , que é o valor retornado pela heurística aplicada correspondente ao valor estimado do caminho de  $n$  até o nó objetivo (DOS ANJOS, 2013).



**Figura 05 – Representação de um Algoritmo de Busca em A\***



Fonte: <https://algorithmicthoughts.wordpress.com/2012/12/15/artificial-intelligence-uniform-cost-searchucs/searchproblem-4/>

### 3.4 Sistemas baseados em regras

É um tipo de mecanismo de busca de soluções, através de um sistema de regras que tem a forma “Condição → Ação”. Condições especificadas ativam regras que por sua vez fazem com que determinadas ações sejam tomadas caso a regra esteja ativada (DALMAU, 2004 *apud* KISHIMOTO, 2004). Também chamado de sistema especialista, os sistemas baseados em regras possuem 2 elementos: (1) um conjunto de regras que representa a base do conhecimento e (2) um mecanismo que processa essas regras de acordo com as informações disponíveis.

Geralmente as regras estão estruturadas em cláusulas “se... então”, a serem processadas uma a uma pelo mecanismo de execução, e buscam imitar a forma natural em que se dá ao raciocínio humano (RIBEIRO *et al*, [2013?]).

Vamos tomar como exemplo o possível comportamento de um gato (NPC) em um jogo:

REGRA 1)	Se	está com sono	e	está entediado	então	dormir
REGRA 2)	Se	está entediado	e	está com fome	então	caçar pássaros
REGRA 3)	Se	está entediado	e	não está com fome	então	chamar atenção do humano
REGRA 4)	Se	não está entediado	e	está com fome	então	comer a ração da tigela
REGRA 5)	Se	não está entediado	e	não está com sono	e	não está com fome
					então	perambular por aí

As regras podem ser processadas de duas maneiras:

- Forma direta ou *Forward Chaining*: a base de regras é percorrida pelo mecanismo de processamento elaborando questões que buscam uma conclusão de acordo com o abastecimento das informações. Também chamamos de *data driven*, os dados fornecidos estabelecem quais regras são ativadas. No exemplo:

QUESTÃO) O gato está com sono?  
 RESPOSTA) Não.  
 CONCLUSÃO) Então ele não deve estar dormindo.  
 QUESTÃO) O gato está entediado?

RESPOSTA)	Sim.
CONCLUSÃO)	Então ele deve procurar algo que o mantenha entretido.
QUESTÃO)	O gato está com fome?
RESPOSTA)	Sim.
CONCLUSÃO)	Então ele deve procurar comida de maneira que isso o entretenha.

- Forma reversa ou *Backward Chaining*: ao invés de questões, primeiro o objetivo no qual se quer chegar é evidenciado, para depois o mecanismo de processamento de regras definir quais os critérios necessários para atingi-lo. Também é chamado de *goal driven*: as regras são examinadas para verificar se atinge o propósito.

OBJETIVO)	Gato deve comer comida da tigela.
RESP. 1)	O gato deve estar com fome.
RESP. 2)	O gato não deve estar acordado.
RESP. 3)	O gato não deve estar entediado.

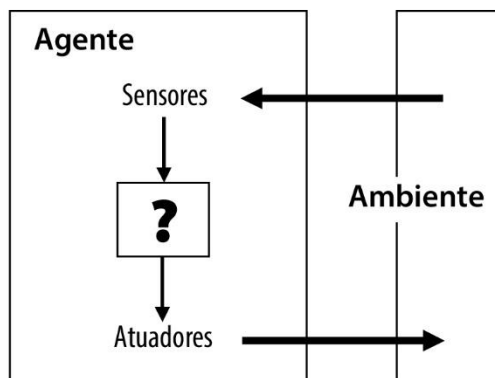
Esses sistemas podem empregar centenas ou milhares de regras encadeadas que demandam um processamento computacional intenso para formar estruturas de conhecimento (RIBEIRO *et al*, 2013). Diferem-se das máquinas de estados finitos por possuírem um tipo de comportamento mais global, se comportando com um conjunto de prioridades e regras.

### 3.5 Agentes inteligentes e Sistemas Multi-Agentes

Agentes são considerados entidades dotadas de sensores que percebem um ambiente e agem neste ambiente através de atuadores. Essas entidades são programas de computador com controle autônomo, que além de perceber e atuar em um ambiente, também são capazes de se adaptar às mudanças; possuem objetivos a alcançar e capacidades que determinam aquilo que conseguem fazer. Podem ajudar na modelagem de NPCs, e no caso dos sistemas multi-agentes, podem desempenhar comportamentos complexos em grupo e que envolvam trabalho em equipe.

Nos sistemas multi-agentes, cada agente interage entre si e com um mesmo ambiente, de forma que consigam executar tarefas complexas demais para um agente único. Podemos encontrar aplicações desses sistemas em jogos de tiro em primeira pessoa ou grupos de entidade em jogos de estratégia (RIBEIRO *et al*, 2013).

**Figura 06 – Exemplo de função de pertinência e variáveis linguísticas da lógica *fuzzy***



Fonte: Adaptado de Ribeiro *et al*, (2013).

### 3.6 Máquinas de Estado Finitos (*Finite State Machines* – FSM)

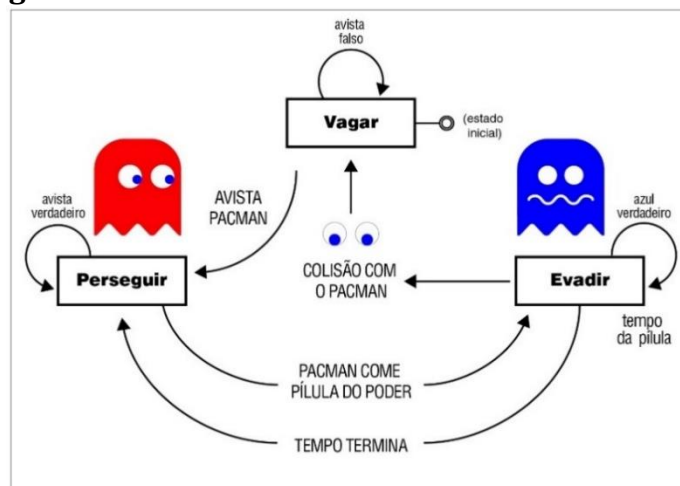
Esse método consiste em uma máquina abstrata que define estados em que um personagem pode se encontrar e quando ele deve mudar de estado; seu estado atual determina como ele deverá agir no jogo. Também é um exemplo de técnica utilizada desde os primeiros jogos eletrônicos que implementavam alguma IA. Ainda hoje muito utilizada, por ser um método de fácil entendimento, implementação e depuração de erros (KISHIMOTO, 2004, p. 5), além de ser um sistema bastante flexível, com a adição de novos estados podendo ser feita com facilidade.

Em uma FSM, temos:

- Um número finito de estados;
- Somente um estado ativo por vez;
- O estado atual deve determinar o comportamento de um personagem ou objeto do jogo;
- Transições entre um estado e outro: se determinada condição é atingida, muda-se de estado.

No contexto de jogos, as máquinas de estados decompõem comportamentos de um objeto ou personagem em estados facilmente gerenciáveis. Um exemplo clássico do funcionamento das FSMs nos jogos é o comportamento dos fantasmas em *Pac Man* – na imagem abaixo, é possível ver que a máquina de estados finitos determina um comportamento de perseguição e evasão, dependendo do estado em que os fantasmas se encontram.

**Figura 7 – Estados Finitos dos Fantasmas - Pacman**



Fonte: Os autores

Máquinas de estado finito possuem limitações, gerando ações repetitivas dependendo da condição que foi atendida, tornando as ações previsíveis ao jogador. Redes neurais, lógica *fuzzy* e a Busca em Árvore Monte Carlo são técnicas de IA que podem ser combinadas com as FSMs para evitar esse problema. Porém, a despeito das suas vantagens, ambientes complexos poderiam demandar um número muito grande de estados.

### 3.7 Lógica *Fuzzy* ou Nebulosa

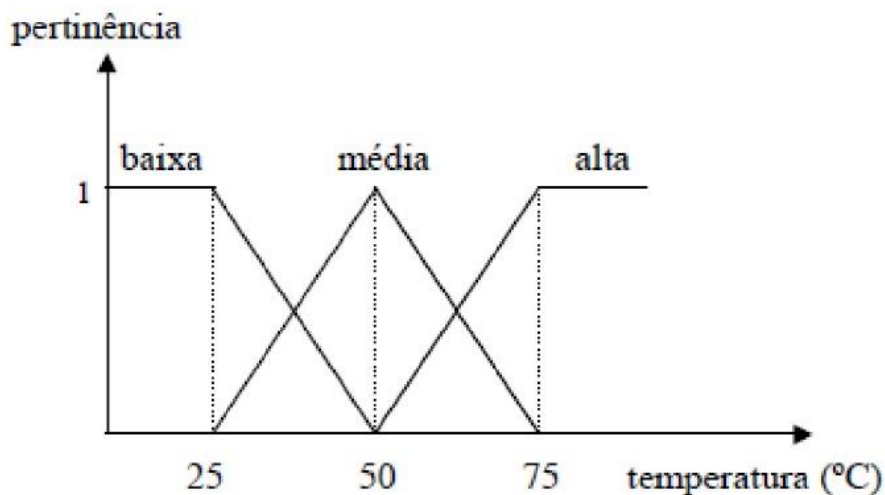
É um tipo de lógica baseada em níveis de incerteza e verdades em uma sentença (WOODSTOCK, 1999 e BOURG, 2004 *apud* KISHIMOTO, 2004), embasada na teoria dos conjuntos nebulosos (*fuzzy*).

O valor verdade de uma proposição é expresso linguisticamente (e.g.: verdade, muito verdade, não verdade, falso, muito falso etc.) e pode ser um subconjunto *fuzzy* de qualquer conjunto parcialmente ordenado, em que cada termo linguístico é interpretado como um subconjunto *fuzzy* do intervalo unitário. Essa técnica é usada para se manusear informações qualitativas de maneira rigorosa, considerando o modo como inexatidões e incertezas são descritas, constituindo formas poderosas de se manipular o conhecimento de maneira conveniente (RIBEIRO *et al*, 2013).

Na lógica *fuzzy*, as variáveis linguísticas são elementos que descrevem informações de maneira geralmente qualitativa, e sua principal função é tentar caracterizar fenômenos complexos ou mal definidos, permitindo tratar sistemas muito complexos de serem analisados matematicamente; as funções de pertinência são formas de representá-la e podem ser definidas de forma padrão, como triangulares, trapezoidais ou Gaussianas ou através da experiência e perspectiva do usuário; o processo de inferência *fuzzy* permite mapear o conhecimento a respeito de um sistema através das regras “Se (condição) → antecedente, Então (ação) → consequente”; e o controlador *fuzzy* prevê uma interface de *fuzzificação*, uma

base de conhecimento, uma base de dados, um procedimento de inferência e uma interface de *fuzzificação* (RIBEIRO *et al*, 2013).

**Figura 08 – Exemplo de função de pertinência e variáveis linguísticas da lógica *fuzzy***



Fonte: Ribeiro *et al*, (2013).

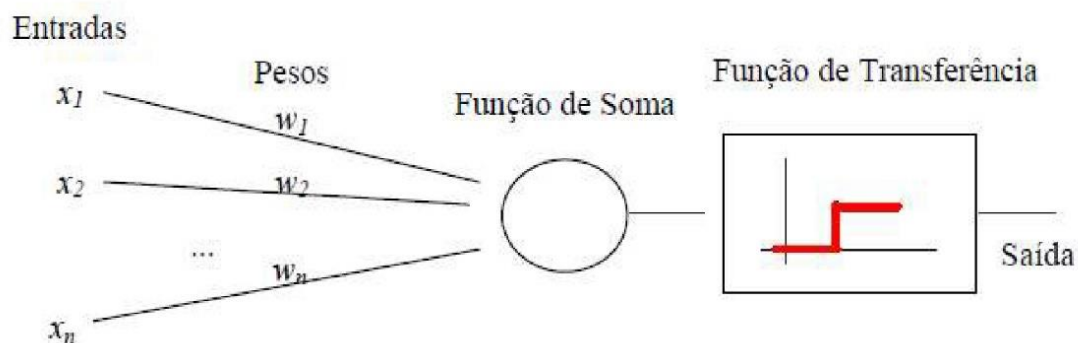
Essa técnica pode ser usada em conjunto com as máquinas de estado, criando máquinas de estado *fuzzy* com o objetivo de criar resultados de ações não tão previsíveis e assim reduzir a quantidade de regras “se – então” (WOODSTOCK, 1999 e BOURG, 2004 *apud* KISHIMOTO, 2004). Um jogo que se utiliza desse tipo de técnica é o *The Sims* (MAXIS, 2000).

### 3.8 Redes Neurais

Essa técnica se inspira no funcionamento dos neurônios do cérebro humano para criar um modelo matemático. Uma rede neural artificial (RNA) é um modelo formado por circuitos compostos por unidades simples interconectadas entre si, capazes de processamento, que são chamadas de neurônios artificiais. Esses neurônios, portanto, simulam as conexões sinápticas biológicas usando uma variável chamada peso. De acordo com Ribeiro *et al* ([2013?]), uma função de soma acumula os dados recebidos através de outros neurônios e uma função de transferência (ou função de ativação) irá processar a função soma e transformá-la; ou seja, um neurônio corresponde a uma soma ponderada de entradas que será aplicada à função de transferência que irá determinar sua ativação.

**Figura 09 – Rede neural artificial**





Fonte: RIBEIRO *et al* ( 2013).

As redes neurais podem apresentar três camadas: uma camada de entrada, uma camada intermediária (que é opcional) e uma camada de saída. A arquitetura da rede neural irá depender de como essas camadas são estruturadas: uma camada de entrada e uma camada de saída com o sinal propagando da entrada para a saída é chamada de *rede feedforward de uma única camada*; quando a camada intermediária está presente e é utilizada como entrada para a camada seguinte, a arquitetura chama-se *rede feedforward de múltiplas camadas*; finalmente, as camadas podem possuir um laço realimentando a saída dos neurônios para outros neurônios da rede neural, formando uma conexão cíclica chamada de *rede recorrente*.

Em um jogo, pode ser aplicada de maneira que personagens “aprendam” de acordo com as escolhas do jogador, o que representa uma particularidade interessante desse tipo de técnica, também conhecida como *machine learning*, ou aprendizado de máquina. Esse aprendizado significa que os parâmetros são ajustados a partir dos estímulos ambientais, que representam os chamados dados de entrada ou treinamento, de maneira que: Estímulo → Adaptação → Novo comportamento da rede neural.

Quando se trata do “aprendizado” observado nas redes neurais ou nas técnicas de *machine learning*, existem três padrões possíveis. O primeiro deles é o próprio aprendizado por reforço, não havendo um supervisor ou modelo de ambiente que interaja com os neurônios fornecendo “respostas” corretas ou comportamentos esperados. Por outro lado, no aprendizado supervisionado, um “supervisor” conhece o ambiente em que atuará o neurônio, fornecendo exemplos de entrada-resposta desejadas, numa espécie de aprendizagem por correção e erro. Outra maneira desse aprendizado ocorrer é com os neurônios competindo entre si, sobrando o neurônio vencedor da competição, o que chamamos de aprendizado não-supervisionado, uma vez que o desempenho da rede não é avaliado e nenhuma medida de erro ou reforço é utilizada para realimentá-la (RIBEIRO *et al*, 2013).

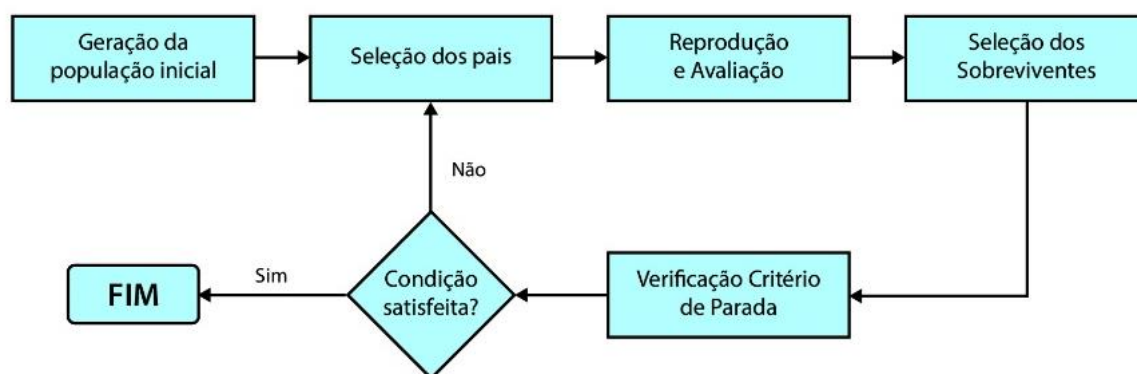
Na prática, a técnica de IA que chamamos de aprendizado por reforço funciona da seguinte maneira: damos um vídeo game e um controlador para um computador. O sistema, apesar de não saber exatamente o que fazer, saberá que existem algumas ações que ele é capaz de realizar. Se o computador decidir se mover para frente, o ambiente o dará algum tipo de recompensa. Se ele decidir se mover para a direita, não recebe recompensas e morre, ou ainda recebe algum tipo de punição. A partir disso, a máquina é capaz de entender que mover-se para

frente é melhor que mover-se para a direita, de forma que o objetivo do algoritmo seja maximizar a recompensa que receberá por suas ações (LUZGIN, 2019).

### 3.9 Algoritmos Evolutivos (AE)

Técnica de Inteligência Artificial que também é baseada na natureza humana, mais especificamente na biologia. Tomando como exemplo o funcionamento da Teoria da Evolução de Darwin, diz-se que uma população de indivíduos é capaz de evoluir, através de mecanismos biológicos como mutação, recombinação, seleção natural e sobrevivência dos indivíduos mais bem adaptados (OLIVEIRA, 2018).

**Figura 10 – Fluxograma de um AE típico**



Fonte: Adaptado de Delbem, 2018 *apud* Oliveira, 2018.

Copiando a biologia, um algoritmo evolutivo é composto por cromossomos, codificados em um conjunto de símbolos chamados genes com diferentes valores (alelos). Dada uma população inicial de indivíduos, geradas aleatoriamente na primeira geração, é aplicada uma função que verifica a aptidão (ou *fitness*) ao ambiente ou situação de cada um dos indivíduos. Os cromossomos que apresentaram a melhor aptidão de acordo com essa função são selecionados para se reproduzirem e compor a geração seguinte. Nesse processo, ocorrem recombinações (troca de alelos entre os indivíduos, chamado crossover) e mutações dos genes, com novas soluções sendo geradas e competindo com as soluções anteriores, mas de maneira que as melhores soluções de cada geração permaneçam na população da geração posterior, até que certa condição de parada seja alcançada. Existem métodos diversos para determinar mutações, seleção dos indivíduos de cada geração e recombinação dos genes de cada indivíduo.

Esse método de IA também pode ser chamado de algoritmo genético, e é comum encontrar aplicações em jogos de tabuleiro, jogos de estratégia e para a criação de personagens, como no jogo *Spore* (MAXIS, 2018). Outra aplicação para algoritmos evolutivos, juntamente com a programação de planejamento e respostas, é na criação de conteúdos de forma procedural (PCGs), que são capazes de automaticamente criar níveis, mapas, histórias, itens, geometria e basicamente qualquer outro tipo de conteúdo.

### 3.10 Flocking

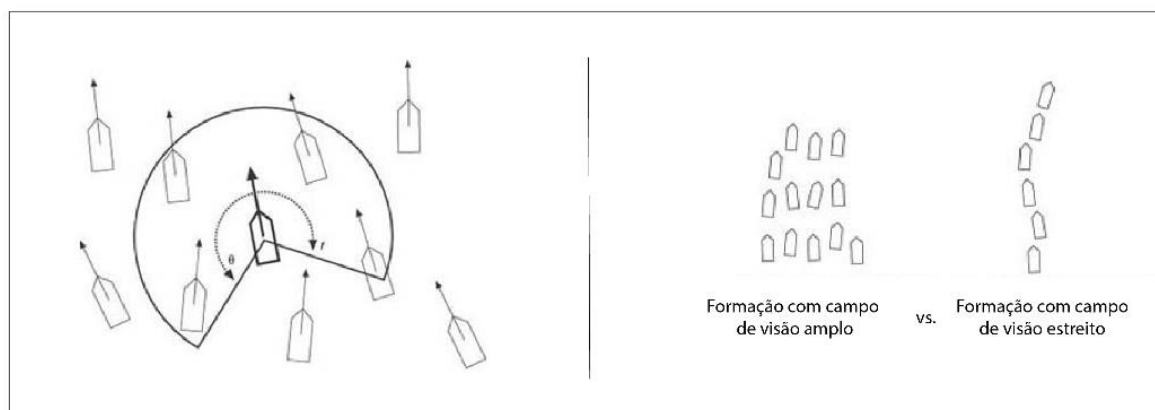
São algoritmos que simulam o movimento de grupos. A palavra *flocking*, traduzida do inglês, significa afluir ou juntar-se, e é usada como o equivalente aos substantivos coletivos em português. Muitas vezes, é mais realista que um conjunto de personagens se mova em grupos coesos, como é o caso de pássaros, rebanhos, cardumes e outros animais gregários. O uso de *flocking* também pode ser observado em movimentação de unidades em jogos de estratégia em tempo real, por exemplo, para controlar a movimentação de unidades de humanos, *trolls*, *orcs*, ou veículos mecanizados de forma geral, como esquadrões de aeronaves em jogos que simulem combate aéreo (SEEMAN e BOURG, 2004).

No desenvolvimento de algoritmos de *flocking*, para que esse comportamento seja satisfatório, três regras devem ser observadas:

1. Coesão: cada unidade deve se guiar de maneira que fique a uma distância média das unidades vizinhas.
2. Alinhamento: cada unidade deve se guiar de modo a se alinhar ao rumo médio de suas vizinhas.
3. Separação: cada unidade deve se guiar de maneira a evitar colisões com as vizinhas.

Dessa maneira, cada unidade deve ser ciente dos seus arredores, sabendo a localização média, a direção e a separação entre si própria e as outras unidades do grupo pelas quais se rodeia.

**Figura 11 – Representação do campo de visão e tipo de formação das unidades**



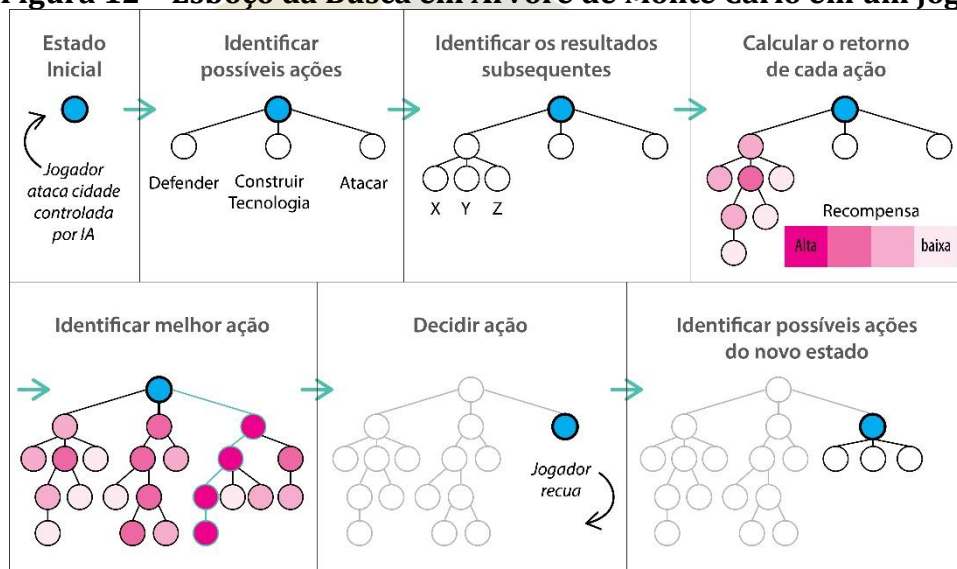
Fonte: Adaptado de Seeman e Bourg, 2004.

### 3.11 Árvores de Decisão

Considerados algoritmos de busca (como citado, busca é um dos problemas mais básicos de IA), as árvores de decisão também são conhecidas como sistemas de busca em árvore. Muitos algoritmos antigos e conhecidos são árvores de decisão, e podem servir como meios para a implementação de outras técnicas de IA. Por exemplo, o algoritmo de MINIMAX e a técnica de algoritmos de busca citados em *path finding* – que é um tipo de busca em árvore, sendo o algoritmo A\* um dos exemplos mais proeminentes.

Uma grande desvantagem das árvores de busca ou decisão é que este tipo de técnica geralmente não suporta mais de dois jogadores (LUCAS *et al*, 2012). É bastante usada em jogos de tabuleiros e outros jogos que precisem verificar todas as probabilidades de jogadas diferentes. Uma inovação recente é o algoritmo MCTS (*Monte Carlo Tree Search*, ou Busca em Árvore Monte Carlo), criado há alguns anos, em 2006, com o foco em buscas intensas: esse algoritmo simula as consequências de sequências longas de ações ao fazer estatísticas de ações aleatórias: através de uma seleção randômica de possíveis ações, há a criação das árvores de decisão a partir dessas ações escolhidas, o que simplifica e aumenta a velocidade dos cálculos que o computador terá que realizar, além de favorecer sua análise sobre qual ação garante uma maior possibilidade de recompensa positiva. Atualmente é o algoritmo que apresenta a melhor performance para jogos de tabuleiro com muitos fatores de ramificação, como o jogo de tabuleiro japonês *Go* (YANNAKAKIS e TOGELIUS, 2015; MAAS, 2019).

**Figura 12 – Esboço da Busca em Árvore de Monte Carlo em um jogo**



Fonte: Adaptado de Lou (2017) *apud* Maas (2019).

O algoritmo MCTS também pode ser usado para criação de personagens controlados por Inteligência Artificial que possuam atitudes bastante verossímeis, além de ser um método útil para a geração de conteúdos de um jogo e no próprio processo de *game design*, sendo capazes de gerar narrativas de maneira automática, para criar ambientes virtuais e gerar novas linhas de roteiro dinamicamente (SIRONI, 2019).

#### 4 Contribuições e vantagens da ia desenvolvida em jogos eletrônicos

Como já foi citado anteriormente neste artigo, os jogos digitais são ótimas plataformas de teste para Inteligência Artificial. O jogo de xadrez, a “drosófila da IA”, atualmente pode ser jogado por qualquer programa de computador relativamente simples, o que implica que, para isso, não há necessidade de ser genuinamente inteligente: os algoritmos criados para se jogar

xadrez só precisam “saber” uma modesta variedade de habilidades do pensamento humano, já que esses jogos se limitam a movimentos baseados em turnos, em um grid de poucas peças com comportamentos determinísticos muito bem definidos (TOGELIUS, 2016).

Por outro lado, temos os jogos eletrônicos. Grande parte da sua popularidade vem do fato de que eles envolvem vários sentidos e habilidades cognitivas humanas. “Vídeo games são uma ótima maneira de treinar algoritmos de IA porque eles são desenhados para conceder à mente humana uma progressão gradual em desafios cada vez mais difíceis” (LUZGIN, 2019, *tradução nossa*).<sup>4</sup>

Conforme já citado, apesar da robótica também ser uma área de aplicação da Inteligência Artificial, os jogos possuem muitas vantagens em relação aos robôs. Esses últimos são lentos, podem ter avarias nas peças ao interagir com o ambiente, ou apresentar superaquecimento de seus componentes à medida que o processamento é demandado. Já os jogos digitais são mais rápidos e seguros, em virtude de não necessitarem um ambiente real para treinar uma IA; segundo Luzgin (2019), enquanto um robô consegue realizar cerca de 10 tarefas para obter algum *feedback* útil, um algoritmo é capaz de jogar um videogame 10 mil vezes e gerar dados valiosos.

Com o uso de técnicas de redes neurais e *deep learning* (aprendizagem profunda), houve alguns avanços significativos nos últimos anos em matéria de IAs capazes de identificar objetos em imagens, entender falas, fazer correspondências entre rostos e nomes, traduzir textos e outras tarefas do tipo, ou seja, o mesmo trabalho dos sistemas sensoriais do nosso cérebro: visão, audição, tato e assim por diante. O que não significa que isso seja tudo o que inteligência é: apesar de ser uma parte importante da inteligência, é apenas uma peça de um sistema que está constantemente trabalhando para decidir o que fazer em seguida (TOGELIUS, 2016).

Por sua vez, os jogos digitais despertam motivações semelhantes às que experimentamos na nossa vida. Jogos clássicos como *Mario* (*Nintendo*), em que o jogador tem um objetivo a conquistar, decisões a serem tomadas, direções a se seguir e ações a serem realizadas, são um exemplo disso. Dependendo das ações que o jogador (ou a IA) efetuar, estará mais próximo de seu objetivo; em contrapartida, se o herói morrer, isso é tratado como um *feedback* negativo, portanto deverá haver uma alteração da estratégia sobre quais dessas decisões tomar (LUZGIN, 2019). Jogos como *StarCraft* (*Blizzard*, 1998) introduzem demandas sobre lidar com informações incompletas, *Skyrim* (*Bethesda*, 2011) necessita que o jogador compreenda narrativas, já a série de jogos *Civilization* (*Electronic Arts*, entre outros) que considere planejamentos a prazos muito longos, como na (TOGELIUS, 2016).

Um exemplo prático bastante interessante de como a IA aplicada em um jogo, e no caso um jogo comercial, é capaz de ajudar em tarefas ou testes de IA em um mundo real, é o aprendizado por reforço encontrado na série de jogos *Grand Theft Auto* (*Rockstar Games*). Como regras de trânsito, rodovias e física dos carros foram usadas no código do jogo e representadas de maneira bem realista, pesquisadores utilizaram esse ambiente para testar algoritmos para treinar carros autônomos (LUZGIN, 2019).

---

<sup>4</sup> “Video games are a great way of training AI algorithms because they are designed to give human minds gradual progression into harder and harder challenges.”



*Open AI*, um laboratório de pesquisas sobre Inteligência Artificial, criou um algoritmo cujo propósito simplesmente era explorar, através de um senso de curiosidade natural, com um sistema de recompensas focado em recompensar conforme se avançava no jogo. O resultado disso foram 11 níveis conquistados no *Super Mario Bros.*, simplesmente movidos por curiosidade, algo muito semelhante a um humano jogando pela primeira vez (LUZGIN, 2019 *apud* MAAS, 2019).

Segundo Togelius (2016), é provável que a nova geração de Inteligências Artificiais, que realmente possamos considerar “inteligentes”, seja uma combinação de métodos como redes neurais, algoritmos evolutivos e árvores de decisão, principalmente a árvore de decisão de Monte Carlo. De acordo com o autor, a coisa mais importante que a humanidade poderia fazer atualmente é inventar uma Inteligência Artificial verdadeira: máquinas ou *softwares* capazes de pensar e agir em situações diversas, de maneira independente. Assim que obtivermos essa inteligência artificial, ela será capaz de resolver todo tipo de outros problemas.

### **Considerações finais**

Conforme é possível observar nas pesquisas realizadas para compor este artigo, a Inteligência Artificial evoluiu muito desde a década de 1950, acompanhando o próprio avanço dos jogos digitais. O esperado com a IA desenvolvida para *games* não é que ela seja imbatível em sua performance, mas sim que ela proporcione a melhor diversão para os jogadores interagirem com ela ou competirem contra ela.

Novas demandas e necessidades para métodos de IA manifestaram-se, e com isso novos e mais complexos algoritmos foram criados, aumentando discussões e pesquisas sobre essa área. Entretanto, ainda há muito para evoluir.

Sob um ponto de vista filosófico, a inteligência humana ainda não foi totalmente bem-sucedida na tarefa de criar uma Inteligência Artificial forte, que além de interagir de maneira lógica e apresentar um comportamento aparentemente inteligente, possua consciência, entenda e apresente uma motivação intrínseca e independente para a realização de suas tarefas.

Como os jogos são boas referências e plataformas de teste para técnicas de IA, eles podem ajudar a evoluir essa inteligência, já que são ambientes relativamente acessíveis e que muitas vezes retratam situações do mundo real. Porém, uma vez que as empresas desenvolvedoras dos jogos digitais comerciais não costumam divulgar quais técnicas foram utilizadas e de que forma a IA foi empregada em seus jogos, a pesquisa acadêmica sobre Inteligência Artificial também se faz importante; os resultados da aplicação de IA em jogos tem potencial para ajudar no desenvolvimento e experimentação de técnicas de Inteligência Artificial em variadas áreas do conhecimento.

### **Referências bibliográficas**

BARBOSA, S. T., VEIGA, J., CARVALHO C. V. **Estudo do Uso de Técnicas de Inteligência Artificial em Jogos 2D**. Revista Eletrônica Teccen, v. 5, n. 1, p. 5 – 20, *Online*, 2012. Disponível

em: <<http://editora.universidadevassouras.edu.br/index.php/TECCEN/article/view/476>>. Acesso em: 15.mai.2021.

COPELAND, B. J. **Artificial Intelligence**. *Encyclopedia Britannica*, 2020. Disponível em: <<https://www.britannica.com/technology/artificial-intelligence>>. Acesso em: 30.mai.2021.

DOS ANJOS, L. R. **ALGORITMOS DE BUSCA DE CAMINHO EM TILE GAMES**. 2013. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade Federal do Pampa, Alegrete, 2013.

FLAUSINO, R. **Estado da Arte da Inteligência Artificial para jogos eletrônicos**. Disponível em: <<https://selectgame.gamehall.com.br/estado-da-arte-da-inteligencia-artificial-para-jogos-eletronicos/>>. Acesso em: 15.mai.2021.

FRANCHI, S. **Chess, Games, and Flies**. *Essays in Philosophy*, v. 6, n. 1, p. 85 - 114, *Online*, 2005. Disponível em: <<https://doi.org/10.5840/eip20056119>>. Acesso em: 19.mai.2021.

FUNGE, J., MILLINGTON I. **Artificial intelligence for games**. 2. ed. [S. l.]: CRC Press, 2009.

IBM Cloud Education. **Artificial Intelligence (AI)**. Disponível em: <<https://www.ibm.com/cloud/learn/what-is-artificial-intelligence>>. Acesso em: 30.mai.2021.

KISHIMOTO, A. **Inteligência Artificial em Jogos Eletrônicos**. 2004. Disponível em: <[https://www.academia.edu/2231411/Inteligência\\_Artificial\\_em\\_Jogos\\_Eletrônicos](https://www.academia.edu/2231411/Inteligência_Artificial_em_Jogos_Eletrônicos)>. Acesso em 12.mai.2021.

LOWOOD, H. E. **Electronic game**. *Encyclopedia Britannica*, 2020. Disponível em: <<https://www.britannica.com/topic/electronic-game>>. Acesso em: 30.mai.2021.

LUCAS, S. M., MATEAS, M., PREUSS, M., SPRONCK, P., TOGELIUS, J. **Artificial and Computational Intelligence in Games (Dagstuhl Seminar 12191)**. *Dagstuhl Reports*, Dagstuhl, v. 2, n. 5, 13 ago 2012. Disponível em: <<https://drops.dagstuhl.de/opus/volltexte/2012/3651/>>. Acesso em: 17.mai.2021.

LUZGIN, R. **Video Games as a Perfect Playground for Artificial Intelligence**. 2019. Disponível em: <<https://towardsdatascience.com/video-games-as-a-perfect-playground-for-artificial-intelligence-3b4ebee36ce>>. Acesso em: 01.jun.2021.

MAAS, L. E. S. **Artificial Intelligence in Video Games**. 2019. Disponível em: <<https://towardsdatascience.com/artificial-intelligence-in-video-games-3e2566d59c22>>. Acesso em 02.jun.2021

OLIVEIRA, G. P. **Métodos de Inteligência Artificial aplicados em jogos baseados em turnos**. 2018. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Faculdade de Computação, Universidade Federal de Uberlândia, Uberlândia, 2018.

RIBEIRO, B., LUCHESE F., ROCHA, M., FIGUEIREDO, V. **Inteligência Artificial em Jogos Digitais**. FEEC/Universidade Estadual de Campinas, 2013. Disponível em:

<<https://www.dca.fee.unicamp.br/~martino/disciplinas/ia369/trabalhos/t4g3>>. Acesso em: 14.mai.2021.

SEEMAN, G., BOURG, D. M. **AI for Game Developers**. 1. ed. [S. l.]: O'Reilly Media, California, 2004.

SIRONI, C. F. **Monte-Carlo Tree Search for Artificial General Intelligence in Games**. Proefschriftmaken.nl || Uitgeverij BOXPress, Maastricht University, Maastricht, 2019. Disponível em: <<https://doi.org/10.26481/dis.20191113cs>>. Acesso em: 15.mai.2021.

TOGELIUS, J. **AI Researchers, Video Games Are Your Friends!**. 2015 7th International Joint Conference on Computational Intelligence (IJCCI), v. 669, p. 3 - 18, 2016. Disponível em: <<https://ui.adsabs.harvard.edu/abs/2016arXiv161201608T>>. Acesso em: 19.mai.2021.

Wikibooks. **Game Creation with XNA/AI/Artificial Intelligence in Games**. Disponível em: <[https://en.wikibooks.org/wiki/Game\\_Creation\\_with\\_XNA/AI/Artificial\\_Intelligence\\_in\\_Games](https://en.wikibooks.org/wiki/Game_Creation_with_XNA/AI/Artificial_Intelligence_in_Games)>. Acesso em: 15.mai.2021.

YANNAKAKIS, G., TOGELIUS J. **A Panorama of Artificial and Computational Intelligence in Games**. IEEE Transactions on Computational Intelligence and AI in Games, v. 7, n. 4, p. 317 - 335, [S. l.: s. n.], 2015.

YANNAKAKIS, G., TOGELIUS J. **Artificial Intelligence and Games**. Springer International Publishing, p. 29, 2018. Disponível em: <[https://link.springer.com/chapter/10.1007%2F978-3-319-63519-4\\_2](https://link.springer.com/chapter/10.1007%2F978-3-319-63519-4_2)>. Acesso em: 15.mai.2012.

