

# Algoritmos Genéticos

Guilherme Miranda de Araújo<sup>#1</sup>, Paulo César Pereira Belmont<sup>\*2</sup>

*Departamento de Ciência da Computação, Universidade Federal de Roraima  
Boa Vista, Roraima, Brasil*

<sup>1</sup>guimiranda2003@gmail.com

<sup>2</sup>paulopereira737@hotmail.com

**Abstract**— This paper analyzes the application of genetic algorithms in different contexts, highlighting two main studies. The first is a review of the scientific article "Simulating Nature's Methods of Evolving the Best Design Solution" by Cezary Janikow and Daniel Clair. The second study is an analysis of the video "Artificial Intelligence with Google Dinosaur" by Ivan Seidel. The paper also presents a modified implementation of the Snake game, where neural networks and genetic algorithms are used to optimize the behavior of the snake, allowing it to learn to collect apples efficiently.

**Link do repositório do trabalho no Github:**  
[https://github.com/Paulo-player/PauloGuilherme\\_FinalProject\\_AA\\_RR\\_2024](https://github.com/Paulo-player/PauloGuilherme_FinalProject_AA_RR_2024)

**Keywords**— Genetic Algorithms, Games, Artificial Intelligence

## I. INTRODUÇÃO

Neste trabalho, serão analisados dois estudos que abordam o uso de algoritmos genéticos em diferentes contextos. O primeiro é o artigo científico "Simulating Nature's Methods of Evolving the Best Design Solution", de Cezary Janikow e Daniel Clair. Este estudo investiga os algoritmos genéticos, abordando desde a forma de representação dos dados até as operações empregadas para alcançar a otimização de soluções. Além disso, discute-se as principais estratégias de aplicação e as limitações inerentes a essa abordagem, destacando pontos como a complexidade de implementação e o custo computacional. O segundo estudo é uma análise do vídeo "Inteligência Artificial com Dinossauro da Google", do canal de YouTube de Ivan Seidel, no qual o autor explica a implementação de um algoritmo genético em conjunto com redes neurais artificiais. O vídeo demonstra como esses métodos são aplicados para controlar o famoso dinossauro no jogo do navegador Google Chrome, oferecendo uma visão prática e acessível da combinação de inteligências artificiais com

algoritmos evolutivos para resolver problemas de controle e decisão.

Por último será descrito uma implementação modificada do jogo Snake (popularmente conhecido como "jogo da cobrinha") que utiliza redes neurais e algoritmos genéticos para busca de cobras com resultados ótimos.

Será abordado as modificações feitas no jogo, a rede neural usada, quais dados são usados para a interação com o ambiente e as técnicas utilizadas para otimização.

## II. ANÁLISE DO ARTIGO "SIMULATING NATURE'S METHODS OF EVOLVING THE BEST DESIGN SOLUTION" DE CEZARY JANIKOW E DANIEL CLAIR

O artigo inicia explicando que os algoritmos genéticos são métodos adaptativos de busca que imitam processos naturais, como seleção, herança de informações, mutação aleatória e dinâmicas populacionais. Esses métodos são particularmente eficazes na otimização de parâmetros numéricos, mas sua aplicação se expandiu para uma variedade de problemas complexos.

Os algoritmos genéticos operam em populações de "cromossomos", que representam potenciais soluções para um problema. Cada cromossomo é composto por genes, e sua estrutura (genótipo) é definida pelo usuário. A avaliação da adaptação de cada cromossomo é feita através de uma função de avaliação específica para o problema em questão.

Os dois mecanismos principais que garantem o comportamento adaptativo dos algoritmos genéticos são a pressão seletiva e a herança de informações. A seleção é um processo estocástico onde a chance de sobrevivência de um agente é proporcional

ao seu nível de adaptação. A cooperação entre agentes é promovida através do operador de crossover, que combina informações de dois cromossomos para gerar descendentes potencialmente mais adaptados.

O funcionamento de um algoritmo genético é descrito como um processo iterativo que envolve a geração de uma população inicial, avaliação, seleção e reprodução. O processo continua até que um critério de parada seja atingido, como um número fixo de iterações ou a convergência para uma solução.

O artigo introduz o conceito de "schemata", que são templates de similaridade que ajudam a entender como diferentes cromossomos podem representar soluções semelhantes. A hipótese dos blocos de construção sugere que os algoritmos genéticos constroem soluções mais complexas a partir de combinações de soluções mais simples, promovendo a exploração e a exploração do espaço de busca.

Os algoritmos genéticos são mais eficazes em problemas com grandes espaços de busca, onde métodos tradicionais não são viáveis. Eles têm sido aplicados com sucesso em otimização de parâmetros numéricos, problemas de transporte, e até mesmo em aprendizado de máquina. No entanto, o artigo também destaca que os algoritmos genéticos não são adequados para todos os problemas, especialmente aqueles com espaços de busca pequenos, onde métodos mais simples podem ser mais eficientes.

O artigo conclui que, apesar das limitações, os algoritmos genéticos têm se mostrado promissores em diversas áreas, especialmente quando adaptados para incorporar informações específicas do domínio do problema. A pesquisa contínua e o desenvolvimento de modelos específicos para problemas complexos são essenciais para maximizar a eficácia dos algoritmos genéticos.

### III. ANÁLISE DO VÍDEO "INTELIGÊNCIA ARTIFICIAL COM DINOSSAURO DA GOOGLE" DE IVAN SEIDEL

O objetivo central deste projeto é desenvolver uma inteligência artificial que

possa interagir autonomamente com o jogo do dinossauro do Google Chrome, superando obstáculos e melhorando seu desempenho através de um processo de aprendizado iterativo. A relevância desse projeto se manifesta na capacidade de replicar, em um ambiente controlado e de baixo custo, a complexidade de comportamentos adaptativos, que são comuns em muitas aplicações de inteligência artificial.



Fig. 1 Captura de tela do "T-Rex Game", também conhecido como "Chrome Dino". Este jogo integrado ao navegador Google Chrome aparece quando um usuário tenta realizar uma pesquisa no navegador enquanto está offline.

A metodologia empregada no projeto é dividida em duas abordagens principais: redes neurais e algoritmos genéticos. As redes neurais permitem que a inteligência artificial aprenda a partir das entradas recebidas, que incluem parâmetros como a distância até o próximo obstáculo e a velocidade do dinossauro. Através de camadas de neurônios, a inteligência artificial transforma essas entradas em saídas que determinam a ação a ser tomada (pular ou não).

Os algoritmos genéticos complementam essa abordagem ao implementar um mecanismo de evolução virtual. Através de seleções naturais, as melhores estratégias de jogo são retidas e cruzadas, possibilitando a introdução de mutações que podem gerar novas abordagens. O sistema é projetado para rodar simulações onde a inteligência artificial testará várias dessas combinações, promovendo uma espécie de competição entre diferentes "indivíduos" da população de soluções.

O processo de aprendizado da inteligência artificial é notável pela sua

natureza iterativa. A cada tentativa de jogo, a inteligência artificial coleta dados sobre seu desempenho, avaliando as ações tomadas com base nos resultados obtidos (pontos, distância percorrida, etc.). Ao longo de múltiplas gerações, ocorrem ajustes nas redes neurais e nos parâmetros dos algoritmos genéticos, permitindo ao sistema aprimorar continuamente sua habilidade de evitar obstáculos e melhorar sua pontuação.

A implementação de uma função de recompensa é fundamental nesta etapa, já que as ações bem-sucedidas são recompensadas, enquanto as falhas levam à penalização. Isso simula o processo de aprendizado por reforço, no qual o agente aprende a maximizar um retorno acumulado ao longo do tempo.

Os resultados obtidos pelo projeto demonstram um notável aumento no desempenho da inteligência artificial ao longo do tempo. Em suas primeiras gerações, a inteligência artificial pode ter dificuldades em pular obstáculos com eficiência, mas, conforme o aprendizado se desenvolve, ela se torna habilidosa em reconhecer padrões e prever a melhor ação a ser tomada. Isso também proporciona um ótimo exemplo da aplicação prática de teorias matemáticas e computacionais no ensino de inteligência artificial.

Um ponto de discussão interessante é a comparação entre o desempenho da inteligência artificial e a intuição humana. A análise dos dados obtidos mostra que, embora a inteligência artificial possa eventualmente superar a capacidade humana em termos de pontuação, seu processo de aprendizado é fundamentalmente diferente, baseando-se em dados e análises algorítmicas em vez de instintos e experiências anteriores.

#### IV. APLICAÇÃO DO ALGORITMO GENÉTICO NO JOGO "SNAKE"

Para fins de aplicação do algoritmo genético, para este trabalho, foi implementada uma versão modificada do clássico jogo "snake". O jogo consiste em uma cobra, que se move em 4 direções diferentes. Conforme a cobra se move, sua

fome aumenta, e ao chegar em um dado nível, a cobra morre. Para evitar isso, o jogador deve coletar "maçãs" antes que a cobra fique totalmente com fome. A forma ótima de se jogar é navegar até a maçã mais próxima, percorrendo o caminho mais curto.

O algoritmo genético neste projeto foi implementado para otimizar o comportamento da cobra no jogo de forma autônoma. Ele segue os princípios básicos dos algoritmos genéticos: inicialização, avaliação, seleção, cruzamento (crossover), mutação e repetição por gerações.

#### V. IMPLEMENTAÇÃO

Cada cobra no jogo é controlada por uma rede neural (Brain), e o algoritmo começa criando uma população de várias cobras. Cada uma tem um "cérebro" com pesos de rede neural diferentes.

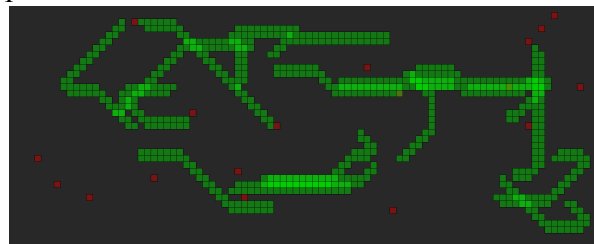


Fig. 2 Captura de tela do jogo Snake com o algoritmo já implementado. Nesse teste foram geradas 50 cobras para serem treinadas

A função de fitness, que avalia o desempenho de cada cobra, é calculada com base na pontuação que ela consegue no jogo (por exemplo, o quanto a cobra sobrevive ou quantas maçãs ela coleta).

Após avaliar todas as cobras, as melhores, com base em sua pontuação, são selecionadas para gerar a próxima geração.

Mutação: A mutação é implementada na função `mutate()` da classe `Brain`. Ela altera os pesos da rede neural de forma aleatória com base em uma taxa de mutação, promovendo variação nas soluções.

Repetição por Gerações: O processo de avaliação, seleção, mutação e criação de uma nova geração se repete até que a solução desejada seja atingida ou um número de gerações seja alcançado.

Essa implementação permite que a cobra "aprenda" com o tempo, otimizando seu comportamento para sobreviver e coletar mais pontos no jogo.

## VI. COMPLEXIDADE DO ALGORITMO GENÉTICO

Consideram-se para esse cálculo:

- P: Tamanho da população
- L: Tamanho do cromossomo
- G: Número de gerações.
- F: Fitness

E assim, os referidos custos para as operações são:

- Inicialização:  $O(P)$
- Fitness:  $O(P * F)$
- Seleção:  $O(P)$
- Crossover:  $O(P * L)$
- Mutação:  $O(P * L)$
- Substituição:  $O(P)$

Somando os custos, temos o seguinte resultado:

$$O(P * F + P + P * L + P * L + P)$$

E ao simplificar, temos a função

$$O(P * F + P * L)$$

$$O(G (P * F + P * L))$$

## REFERÊNCIAS

- [1] C. Z. Janikow and D. St. Clair, "Genetic algorithms. Simulating nature's methods of evolving the best design solution," in IEEE Potentials, vol. 14, no. 1, pp. 31-35, Feb.-March 1995.
- [2] I. Seidel, "Inteligência Artificial com Dinossauro da Google," YouTube, 5-Out-2020. [Online]. Available: <https://youtu.be/P7XHqZjXQs>. [Acessado: 23-Set-2024].
- [3] "T-Rex Game", Wikipedia, [Online]. Available: [https://pt.wikipedia.org/wiki/T-Rex\\_Game](https://pt.wikipedia.org/wiki/T-Rex_Game). [Acessado: 23-Set-2024].
- [4] "Snake (1998 video game)", Wikipedia, [Online]. Available: [https://en.wikipedia.org/wiki/Snake\\_\(1998\\_video\\_game\)](https://en.wikipedia.org/wiki/Snake_(1998_video_game)). [Acessado: 23-Set-2024].