

Trabalho 2 de Inteligência Artificial

Miguel Vieira Machado Pim

Abstract

Este trabalho apresenta o desenvolvimento de um agente inteligente treinado para jogar uma versão simplificada do jogo *Space Invaders*, utilizando técnicas de aprendizado por reforço. O agente toma decisões com base em uma rede neural artificial, cujos pesos foram otimizados por meio da meta-heurística Voo dos Morcegos (*Bat Algorithm*), inspirada no comportamento de eco-localização de micromorcegos.

A rede foi avaliada em 30 execuções do jogo e comparada com outros três agentes: um agente baseado em regras, uma rede neural treinada com algoritmo genético e um jogador humano. Os resultados obtidos mostram que o agente otimizado com o Voo dos Morcegos obteve o melhor desempenho médio, sendo também o único capaz de alcançar pontuações acima de 100 pontos.

Análises estatísticas, como **testes t** e de **Wilcoxon**, confirmaram que seu desempenho foi significativamente superior ao dos demais agentes em quase todas as comparações. Esses achados demonstram a eficácia da abordagem proposta na tarefa de controle de agentes em ambientes dinâmicos.

Keywords: Meta-Heurísticas, Aprendizado por reforço, Redes Neurais

1. Introdução

Este artigo apresenta o segundo trabalho desenvolvido na disciplina de Inteligência Artificial (IA), sob orientação do Prof. Flávio Varejão, no curso de Ciência da Computação da Universidade Federal do Espírito Santo (UFES).

⁵ Algoritmos de meta-heurísticas, como a Otimização por Enxame de Partículas (PSO) e os Algoritmos Genéticos (GA), têm se mostrado ferramentas poderosas

para resolver problemas complexos de otimização em diversos domínios. Esses métodos se destacam pela capacidade de explorar espaços de busca de forma eficiente, mesmo em cenários com funções objetivo não deriváveis ou altamente
10 multimodais.

Neste trabalho, é proposto o uso da meta-heurística **Voo dos Morcegos** (**Bat Algorithm**) para otimizar os pesos de uma rede neural artificial. O objetivo é criar um agente inteligente capaz de jogar uma versão simplificada do jogo *Space Invaders*, utilizando aprendizado por reforço. A rede neural atua
15 como classificador de ações e é treinada a partir das pontuações obtidas durante o jogo, de forma a maximizar seu desempenho.

Ao final do treinamento, o agente é avaliado em 30 execuções consecutivas e comparado com outros métodos: um agente baseado em regras, uma rede neural treinada com algoritmo genético e um jogador humano. A comparação é feita
20 por meio de análises estatísticas, incluindo **testes t** e de **Wilcoxon**, além de representações gráficas como boxplots.

2. Descrição do Classificador

Neste trabalho, o classificador utilizado para a tomada de decisão do agente inteligente é uma rede neural artificial (*RNA*) do tipo *feedforward*, também
25 conhecida como rede neural totalmente conectada. Esse tipo de rede é composto por camadas de neurônios dispostos de forma sequencial, em que cada neurônio de uma camada é conectado a todos os neurônios da camada subsequente. Seu objetivo é aprender, a partir das entradas do ambiente, a escolher a ação mais adequada em cada estado do jogo.

30 A arquitetura adotada foi composta por quatro camadas densamente conectadas: uma camada de entrada com 27 neurônios, duas camadas ocultas com 32 e 16 neurônios, e uma camada de saída com 3 neurônios, correspondentes às ações possíveis no jogo (0: sem ação, 1: mover para cima, 2: mover para baixo). Os 27 neurônios de entrada correspondem à representação do estado
35 extraída da função `get_state`, sendo 25 posições da grade sensorial (5×5) mais

duas variáveis internas (posição vertical e velocidade do jogador).

As funções de ativação utilizadas foram a tangente hiperbólica nas camadas ocultas, escolhida por sua capacidade de modelar relações não lineares e suavizar os gradientes durante o aprendizado. Na camada de saída, foi utilizada a função *softmax*, que transforma os valores dos neurônios de saída em uma distribuição de probabilidades sobre as ações possíveis, permitindo a escolha da ação com maior probabilidade.

3. Descrição da Meta Heurística

A metaheurística adotada neste trabalho é o Voo dos Morcegos (*Bat Algorithm*)[1]. Trata-se de uma técnica inspirada no comportamento de eco-localização dos micromorcegos, que utilizam pulsos ultrassônicos para navegar, evitar obstáculos e localizar presas no escuro. Esse mecanismo natural de detecção é modelado computacionalmente para guiar agentes virtuais em busca de soluções ótimas em espaços de alta dimensionalidade.

3.1. Intuição Biológica

Micro-morcegos emitem pulsos sonoros em frequências que variam de $25kHz$ a $150kHz$, escutando os ecos refletidos para inferir a posição, a velocidade e até mesmo o tipo de alvo detectado. Durante a caça, eles ajustam dinamicamente a frequência dos pulsos, a taxa de emissão e a intensidade sonora, otimizando sua capacidade de localizar presas. Esses comportamentos foram abstraídos em três princípios fundamentais do algoritmo:

- Cada morcego usa eco-localização para detectar distâncias e identificar presas;
- Os morcegos ajustam suas frequências de emissão, velocidades e posições com base na qualidade da solução atual;
- A intensidade do som (*loudness*) diminui à medida que se aproxima da presa, enquanto a taxa de emissão de pulsos (*pulse rate*) aumenta.

3.2. Formulação Computacional

Na formulação do algoritmo, cada *morcego virtual* representa uma solução candidata (neste caso, um vetor de pesos da rede neural). O algoritmo é populacional, ou seja, várias soluções evoluem simultaneamente ao longo das iterações. O comportamento de cada morcego é modelado pelas seguintes variáveis:

- **Posição** x_i e **velocidade** v_i no espaço de busca;
- **Frequência** f_i , associada à escala de movimentação;
- **Loudness** A_i , que representa a disposição do morcego em aceitar novas soluções;
- **Pulse rate** r_i , controlando a probabilidade de executar busca local.

O processo de otimização ocorre iterativamente, e a atualização das soluções segue as seguintes equações:

75 1. **Frequência:**

$$f_i = f_{\min} + (f_{\max} - f_{\min}) \cdot \beta$$

onde $\beta \in [0, 1]$ é um vetor aleatório obtido a partir de uma distribuição uniforme.

2. **Velocidade e posição:**

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_*) f_i$$

$$x_i^t = x_i^{t-1} + v_i^t$$

80 onde x_* representa a melhor solução global conhecida até o momento.

3. **Busca local (executada com probabilidade r_i):**

$$x_{\text{local}} = x_* + \epsilon A_i^t, \quad \epsilon \sim U(-1, 1)$$

4. **Atualização da intensidade sonora e taxa de emissão de pulsos:**

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^0 [1 - e^{-\gamma t}]$$

com $\alpha \in (0, 1)$ e $\gamma > 0$ como parâmetros de decaimento. Para este trabalho foi adotado $\alpha = 0.9$ e $\beta = 0.9$

85 Nesta abordagem, a exploração global do espaço de busca é controlada pelas
frequências f_i , enquanto a intensificação (refinamento local) é realizada com base
nos valores de A_i e r_i . Esse equilíbrio entre exploração e intensificação permite
ao algoritmo convergir para soluções de alta qualidade, mesmo em espaços de
alta dimensionalidade. O Algoritmo 1 mostra o pseudocódigo do funcionamento
90 da meta-heurística.

Algorithm 1: Algoritmo do Voo dos Morcegos (Bat Algorithm)

Input: Função objetivo $f(x)$, limites de busca, número de morcegos n ,

número máximo de iterações T_{\max}

Output: Melhor solução x^* encontrada

```

1 Inicializar população de morcegos  $x_i$ , velocidades  $v_i$  e frequências  $f_i$ 
2 Inicializar loudness  $A_i$  e taxa de emissão de pulsos  $r_i$  para cada
morcego
3 Avaliar  $f(x_i)$  para todos os morcegos e encontrar melhor solução  $x^*$ 
4 for  $t \leftarrow 1$  to  $T_{\max}$  do
5   for cada morcego  $i = 1$  to  $n$  do
6     Gerar nova frequência  $f_i$  a partir de distribuição uniforme
7     Atualizar velocidade:  $v_i \leftarrow v_i + (x_i - x^*) \cdot f_i$ 
8     Atualizar posição:  $x_i \leftarrow x_i + v_i$ 
9     if  $rand > r_i$  then
10       Gerar uma solução local:  $x_i \leftarrow x^* + \epsilon A_i$ 
11     Avaliar  $f(x_i)$ 
12     if  $rand < A_i$  e  $f(x_i) < f(x^*)$  then
13       Aceitar nova solução
14       Atualizar  $A_i$  e  $r_i$ :
15          $A_i \leftarrow \alpha A_i, \quad r_i \leftarrow r_i^0 [1 - \exp(-\gamma t)]$ 
16       Atualizar  $x^*$  se necessário
17 return melhor solução  $x^*$ 

```

4. Resultados

A meta-heurística do Voo dos Morcegos foi executada com um limite máximo de 1000 iterações e um tempo máximo de execução de 12 horas. Durante o processo de treinamento da rede neural, o melhor desempenho obtido foi um score de aproximadamente 122 pontos no jogo. A Figura 1 apresenta a evolução dos melhores scores ao longo das iterações do algoritmo, permitindo observar a progressão do aprendizado.

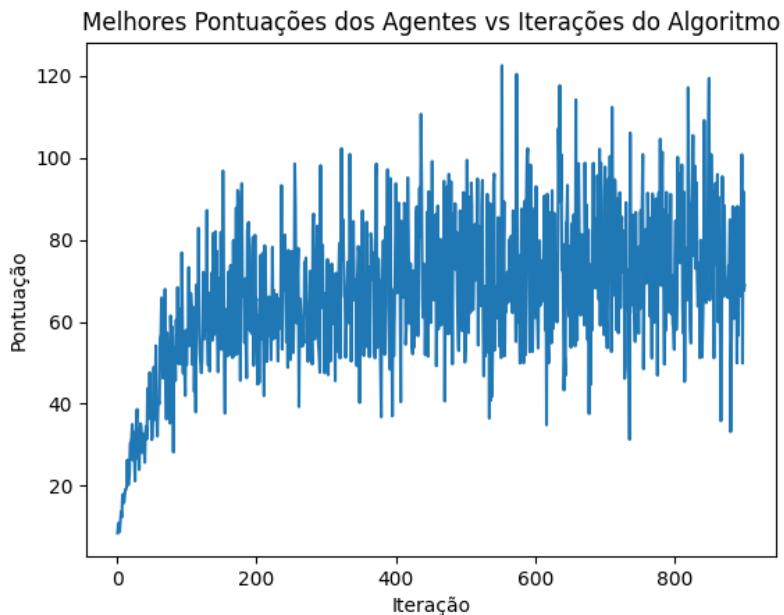


Figure 1: Iteração do algoritmo vs Melhor score

Após a definição do melhor conjunto de pesos durante o treinamento, a rede neural foi submetida a 30 execuções do jogo para avaliação de desempenho. A média e o desvio padrão dos scores obtidos nessas execuções foram calculados para análise estatística. A Tabela 1 compara os resultados do agente treinado com o Voo dos Morcegos com os de outros agentes: um agente baseado em regras, uma rede neural treinada por algoritmo genético e um jogador humano.

Table 1: Resultados dos agentes jogando o jogo.

Agente	Média	Desvio Padrão
Bat	44.78	29.05
Rule Based	13.97	5.67
Genetic	38.99	17.01
Human	22.19	8.11

105 A Figura 2 mostra o boxplot gerado a partir dos resultados de cada um dos agentes. Nela, podemos observar que o agente treinado pelo algoritmo genético foi o melhor.

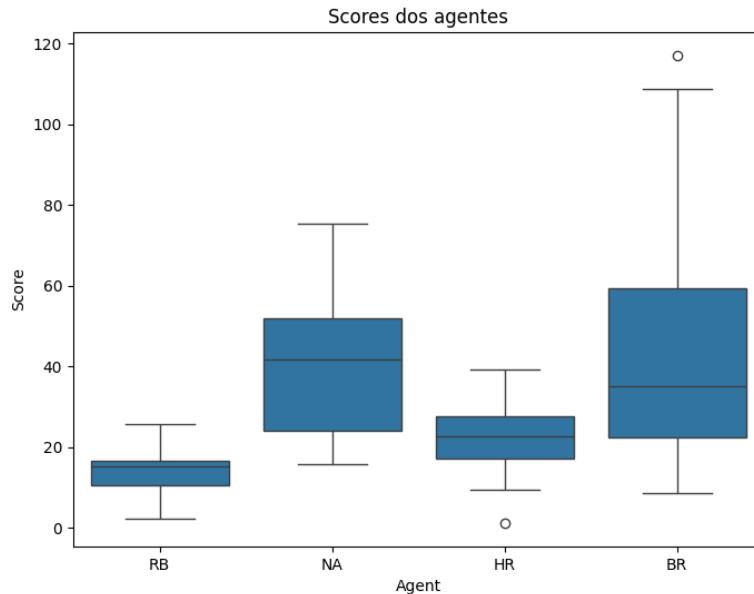


Figure 2: Boxplot dos resultados obtidos por cada agente.

A Tabela 2 mostra os resultados pareados *p-values* comparando o desempenho de cada um dos agentes entre si. A parte superior da matriz (acima da diagonal principal) exibe os valores obtidos pelo teste t pareado com amostras independentes, enquanto a parte inferior apresenta os valores do teste não

paramétrico de **Wilcoxon**. Ambos os testes foram realizados com nível de significância de 5%, permitindo avaliar se as diferenças de desempenho entre os agentes são estatisticamente significativas.

¹¹⁵ Observa-se que, para a maioria das comparações, os *p-values* obtidos são inferiores a 0,05, indicando diferenças estatisticamente significativas entre os agentes. A única exceção ocorre na comparação entre a rede neural treinada com o algoritmo genético e aquela otimizada pelo Voo dos Morcegos. Nessa comparação, os testes estatísticos não apontam diferença significativa, com *p-values* de 0,7766 para o teste t e 0,3582 para o teste de **Wilcoxon**, ambos acima do limiar de 5%.

¹²⁰

Table 2: Resultados pareados *p-values*.

Rule Based	0.0000	0.0002	0.0000
0.0000	Genetic	0.0000	0.7766
0.0000	0.0000	Human	0.0032
0.0000	0.3582	0.0002	Bat

5. Conclusões

5.1. Análise geral dos resultados

Os resultados experimentais evidenciam que o agente treinado com a metaheurística do Voo dos Morcegos apresentou, em média, o melhor desempenho entre todos os agentes avaliados, com uma pontuação média de 44,78 e desvio padrão de 29,05. Esse resultado supera significativamente o desempenho do agente baseado em regras ($13,97 \pm 5,67$), do jogador humano ($22,19 \pm 8,11$) e da rede neural treinada com algoritmo genético ($38,99 \pm 17,01$).

¹²⁵

¹³⁰ A Figura 1 ilustra a curva de aprendizado do agente otimizado pelo Voo dos Morcegos. Observa-se uma tendência clara de crescimento nas melhores pontuações ao longo das iterações, especialmente nas primeiras 200 iterações. Após esse ponto, a curva apresenta oscilações mais intensas, mas mantém um patamar elevado, indicando que o algoritmo é capaz de explorar regiões promissoras do espaço de busca mesmo nas fases finais do treinamento.

¹³⁵

Já a Figura 2 mostra a distribuição das pontuações dos agentes ao longo de 30 execuções. É possível observar que, embora o agente baseado no Voo dos Morcegos apresente maior variabilidade nos resultados, ele também atinge as pontuações mais altas, inclusive com valores acima de 110 pontos — o que não
140 é alcançado por nenhum outro agente. Por outro lado, o agente treinado com o algoritmo genético apresenta uma distribuição mais concentrada e estável, com menor variância e um desempenho médio ligeiramente inferior, o que sugere um comportamento mais consistente, porém com menor capacidade de alcançar pontuações extremas.

145 Por fim, a análise estatística (Tabela 2) confirma que as diferenças de desempenho entre os agentes são estatisticamente significativas na maioria das comparações, com *p-values* inferiores a 0,05 nos testes *t* e de Wilcoxon. A única exceção ocorre entre os agentes otimizados com o algoritmo genético e o Voo dos Morcegos, cuja comparação resultou em *p-values* de 0,7766 para o teste **t**
150 e 0,3582 para o teste de **Wilcoxon**, não indicando diferença estatisticamente significativa entre seus desempenhos médios.

5.2. Contribuições do Trabalho

Este trabalho contribui ao demonstrar a viabilidade do uso da meta-heurística do Voo dos Morcegos na otimização de redes neurais em ambientes de reforço
155 não supervisionado. A formulação do agente neural e sua integração com o ambiente do jogo permitiram testar, de forma prática, a eficácia da estratégia em um cenário dinâmico e com múltiplas possibilidades de decisão.

Além disso, o trabalho apresenta uma comparação experimental rigorosa, com base em testes estatísticos e visualizações adequadas, fornecendo evidências
160 sólidas do desempenho relativo entre diferentes abordagens de treinamento de agentes.

5.3. Melhorias e Trabalhos Futuros

Algumas melhorias podem ser implementadas para aprimorar os resultados obtidos. Entre elas, destaca-se a possibilidade de explorar outras arquiteturas

¹⁶⁵ de rede neural, incluindo o uso de camadas convolucionais ou recorrentes, que podem captar melhor a dinâmica temporal e espacial do jogo.

Em relação à meta-heurística, podem ser avaliadas variações do Voo dos Morcegos com mecanismos adaptativos de parâmetros, estratégias híbridas com outras heurísticas (como PSO ou DE), ou ainda o uso de mecanismos elitistas para preservação das melhores soluções.
¹⁷⁰

Futuramente, pretende-se também aplicar o agente a outros jogos ou tarefas de controle contínuo, de forma a avaliar a generalização do método proposto para diferentes contextos.

References

- ¹⁷⁵ [1] X.-S. Yang, A new metaheuristic bat-inspired algorithm, in: J. R. González, et al. (Eds.), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Vol. 284 of *Studies in Computational Intelligence*, Springer, Berlin, 2010, pp. 65–74.