

# Segundo Trabalho de Inteligência Artificial e Sistemas Inteligentes

Milena Curtinhas Santos

*Universidade Federal do Espírito Santo*

---

## Abstract

Este trabalho propõe a utilização do Algoritmo dos Morcegos (BA) como meta-heurística para treinar uma rede neural aplicada ao jogo *Space Invaders*. O objetivo é desenvolver um agente capaz de aprender estratégias de forma eficiente em um ambiente dinâmico e desafiador. A metodologia combina um classificador baseado em rede neural com a busca orientada pelo BA, explorando seu potencial para otimização em cenários de alta complexidade. Os experimentos realizados demonstraram que a abordagem é capaz de gerar agentes com bom desempenho, evidenciando a viabilidade do uso de meta-heurísticas em problemas de aprendizado de máquina e jogos digitais.

*Keywords:* Inteligência Artificial, Algoritmo dos Morcegos, Meta-heurística, Aprendizado de Máquina, Rede Neural

---

## 1. Introdução

O aprendizado por reforço representa um paradigma fundamental da inteligência artificial que permite a criação de agentes autônomos capazes de aprender através da interação com o ambiente, utilizando recompensas e penalidades para otimizar seu comportamento. Esta abordagem tem demonstrado eficácia notável no desenvolvimento de sistemas inteligentes para jogos eletrônicos, onde o agente deve tomar decisões sequenciais em ambientes dinâmicos e incertos. O trabalho apresentado nesse artigo investiga a aplicação de aprendizado por reforço para desenvolver um agente inteligente capaz de jogar

10 uma versão simplificada do clássico jogo *Space Invaders*, combinando redes neurais artificiais como classificador com a meta-heurística do Voo dos Morcegos [1] para otimização dos parâmetros. O objetivo principal é desenvolver e avaliar um agente inteligente que maximize sua pontuação no jogo através da otimização evolutiva de uma rede neural, comparando seu desempenho com agentes baseados em regras, outros agentes neurais e jogadores humanos.

## 2. Descrição do Classificador

O classificador implementado consiste em uma rede neural artificial *feed-forward* de arquitetura compacta, especificamente projetada para processar informações sensoriais do ambiente de jogo e produzir decisões de movimento em tempo real. A rede neural foi desenvolvida como estrutura multicamadas totalmente conectada com bias, implementada na classe `NeuralNetworkAgent`, otimizada para eficiência computacional e capacidade de generalização.

A arquitetura da rede neural possui configuração  $27+1 \rightarrow 16 \rightarrow 8 \rightarrow 3$ , totalizando 611 pesos sinápticos. A camada de entrada processa 27 valores provenientes dos sensores do jogo, representados por uma grade  $5 \times 5$  posicionada à frente do jogador que detecta obstáculos no ambiente, complementados por duas variáveis internas: a posição Y normalizada do jogador e a velocidade atual do jogo normalizada. O sinal de entrada é expandido para 28 neurônios através da adição de um termo de bias constante (1.0), permitindo maior flexibilidade no ajuste dos limites de decisão da rede.

A primeira camada oculta contém 16 neurônios que recebem as informações preprocessadas dos sensores e variáveis internas, realizando uma transformação não-linear através da função de ativação tangente hiperbólica ( $\tanh$ ). Esta função foi escolhida por suas propriedades de saturação suave e gradientes bem comportados, facilitando o processo de otimização pelos algoritmos evolutivos. A segunda camada oculta, composta por 8 neurônios, atua como um estágio de refinamento das características extraídas pela camada anterior, também utilizando ativação  $\tanh$  e incorporando bias para cada neurônio.

A camada de saída possui 3 neurônios correspondentes às três ações possíveis  
40 do agente: permanecer parado (ação 0), mover para cima (ação 1) e mover para baixo (ação 2). Esta camada utiliza a função de ativação softmax, que normaliza as saídas em uma distribuição de probabilidade, garantindo que a soma das probabilidades das três ações seja igual a 1.0. A decisão final é tomada através da seleção da ação com maior probabilidade.

45 A escolha desta arquitetura compacta foi motivada pela comparação com a configuração  $27+1 \rightarrow 32 \rightarrow 16 \rightarrow 3$  que possuía 1.475 pesos, representando uma redução de 58,6% no espaço de parâmetros. Esta otimização trouxe vantagens significativas: convergência mais rápida do algoritmo dos morcegos devido ao espaço de busca menor, maior eficiência computacional permitindo mais 50 gerações dentro do limite de 12 horas, prevenção de *overfitting* através da regularização implícita da arquitetura menor, e melhor adequação ao problema que possui entrada relativamente simples de 27 dimensões. A progressão em funil ( $16 \rightarrow 8 \rightarrow 3$ ) força a rede a extrair características hierárquicas essenciais, eliminando a redundância desnecessária da configuração original.

55 O processo de inferência segue um fluxo direto, no qual o estado é propagado sequencialmente através das camadas. Para garantir a estabilidade numérica, a implementação incorpora *clipping* nos valores intermediários, limitando-os ao intervalo  $[-500, 500]$ , e aplica a subtração do valor máximo na função *softmax*. A inicialização dos pesos segue o método Xavier/Glorot [2], com intervalo 60  $[-\text{limit}, \text{limit}]$ , onde

$$\text{limit} = \sqrt{\frac{6}{\text{fan\_in} + \text{fan\_out}}}$$

, promovendo gradientes estáveis e facilitando a convergência..

### 3. Descrição da Meta-heurística

A meta-heurística empregada baseia-se no Algoritmo dos Morcegos (Bat Algorithm - BA), proposto por Yang [1], que se inspira no comportamento 65 de ecolocalização dos micromorcegos. O algoritmo modela três características

principais: uso de ecolocalização para detectar presas e obstáculos, ajuste automático da frequência e taxa de emissão de pulsos baseado na proximidade do alvo, e variação da intensidade sonora conforme a aproximação da presa.

O funcionamento baseia-se no movimento de morcegos virtuais em um espaço de busca multidimensional, onde cada morcego representa uma solução candidata correspondente aos 611 pesos da rede neural. Cada morcego possui parâmetros individuais de frequência ( $f_i$ ), velocidade ( $v_i$ ), intensidade sonora ( $A_i$ ) e taxa de emissão de pulsos ( $r_i$ ). A cada iteração, os morcegos atualizam suas posições através de busca global (movimento direcionado à melhor solução conhecida) e busca local (refinamento nas proximidades de soluções promissoras). O algoritmo incorpora convergência natural através da redução gradual da intensidade sonora e aumento da taxa de emissão de pulsos quando soluções melhores são encontradas.

A função fitness no contexto do problema avalia cada configuração de pesos executando o agente neural correspondente em múltiplas partidas do jogo. O morcego que representa o melhor conjunto de pesos (maior score médio) atua como atrator global, influenciando o movimento de toda a população em direção a regiões promissoras do espaço de soluções. A busca local é implementada através de caminhadas aleatórias ao redor das melhores soluções, com amplitude controlada pela intensidade sonora média da população, permitindo ajustes finos nos pesos da rede neural.

A aplicação específica utiliza população de 100 morcegos, cada um codificando os 611 pesos da arquitetura  $27+1 \rightarrow 16 \rightarrow 8 \rightarrow 3$ . A inicialização segue o método Xavier [2] para promover gradientes estáveis. A função fitness avalia cada conjunto de pesos através de 5 execuções do jogo, calculando o score médio como medida de desempenho. Os parâmetros de frequência variam no intervalo  $[0, 1]$ , controlando a intensidade do movimento direcionado, enquanto os limites dos pesos são mantidos em  $[-2.0, 2.0]$  através de clipping, adequado para redes neurais com ativação tanh.

O artigo descreve uma busca local simples, porém, para melhorar os resultados, a busca local foi aprimorada com distribuição gaussiana em lugar da

uniforme original, além do uso de Voo de Lévy para diversificar a busca quando o algoritmo detecta estagnação prolongada, uma técnica usada para escapar de mínimos locais. Além disso, ao contrário da formulação original, que emprega 100 parâmetros fixos, a implementação para o contexto do trabalho adota uma atualização mais dinâmica e adaptativa dos componentes do algoritmo. A intensidade sonora (*Loudness*) mantém-se fiel à equação de referência, mas a taxa de emissão de pulsos (*Pulse Rate*) é atualizada de acordo com um `improvement_factor`, tornando-se responsiva ao progresso da otimização. Outro diferencial é a função 105 `adaptive_parameters`, que ajusta automaticamente `fmax` e  $\alpha$  ao longo das gerações. Essa adaptação estratégica favorece uma exploração ampla nas fases iniciais e uma exploração intensiva nas etapas finais, ampliando a eficácia da meta-heurística.

Esta implementação combina os mecanismos fundamentais do algoritmo dos 110 morcegos com adaptações específicas para otimização de redes neurais, resultando em processo robusto capaz de encontrar configurações eficazes de pesos dentro das restrições temporais estabelecidas, aproveitando tanto a capacidade de exploração global quanto o refinamento local característico do comportamento de ecolocalização dos morcegos.

#### 115 4. Resultados

Os resultados experimentais demonstram o desempenho superior do agente 120 neural otimizado pelo Algoritmo dos Morcegos em comparação com outros métodos. A avaliação foi conduzida através de 30 execuções independentes do jogo para cada agente, garantindo robustez estatística na comparação de desempenho.

O agente desenvolvido com o Algoritmo dos Morcegos alcançou performance notavelmente superior, obtendo score médio de  $70,47 \pm 43,00$ , conforme apresentado na Tabela 2. Este resultado supera significativamente todos os outros agentes avaliados: o agente neural otimizado por algoritmo genético ( $38,99 \pm$  125  $17,01$ ), o agente humano ( $22,18 \pm 8,11$ ) e o agente baseado em regras otimizado

por algoritmo genético ( $13,97 \pm 5,67$ ). O desempenho representa uma melhoria de 80,8% em relação ao agente neural com algoritmo genético e 404,5% em relação ao agente baseado em regras.

Table 1: Resultados dos Testes 1-10 de Desempenho por Agente

Agente	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
Bat	105,78	72,11	16,30	92,76	46,41	110,37	40,75	124,05	106,77	23,24
Regras	12,69	16,65	6,97	2,79	15,94	10,22	21,90	4,35	6,22	9,95
Neural	38,32	54,53	61,16	27,55	16,08	26,00	25,33	18,30	39,76	48,17
Humano	27,34	17,63	39,33	17,44	1,16	24,04	29,21	18,92	25,71	20,05

Table 2: Resultados dos Testes 11-20 de Desempenho por Agente

Agente	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20
Bat	76,33	58,56	17,23	41,81	17,36	36,27	99,85	50,68	16,70	19,99
Regras	19,94	20,56	15,74	17,68	7,16	15,68	2,37	15,43	15,13	22,50
Neural	44,77	47,54	75,43	23,68	16,83	15,81	67,17	53,54	33,59	49,24
Humano	31,88	15,39	22,50	19,27	26,33	23,67	16,82	28,45	12,59	33,01

Table 3: Resultados dos Testes 21-30, Média e Desvio Padrão de Desempenho por Agente

Agente	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	Média	DP
Bat	86,83	116,40	19,28	112,35	124,32	62,51	177,42	110,46	20,88	110,28	<b>70,47</b>	<b>43,00</b>
Regras	25,82	15,85	17,02	16,74	14,69	11,73	13,80	15,13	12,35	16,19	13,97	5,67
Neural	52,65	16,35	44,05	56,59	63,23	43,96	43,82	19,19	28,36	18,65	38,99	17,01
Humano	21,74	14,23	27,90	24,80	11,35	30,12	17,08	22,96	9,41	35,22	22,18	8,11

A análise da distribuição dos resultados, apresentada na Figura 1, revela características importantes do comportamento de cada agente. O agente Bat Algorithm apresenta maior amplitude de scores, com valores mínimos próximos a 16,30 e máximos alcançando 177,42, indicando estratégias mais diversificadas. O primeiro quartil situa-se em aproximadamente 30, a mediana em 67 e o terceiro quartil em 110, demonstrando distribuição assimétrica com concentração de bons desempenhos.

A análise estatística confirma diferenças significativas entre todos os pares de agentes através de testes paramétricos e não-paramétricos, conforme apresentado na Tabela 4. O teste t para amostras independentes resultou em p-values inferiores a 0,001 para todas as comparações envolvendo o agente Bat Algorithm. Os testes de Wilcoxon confirmam estas diferenças com p-values igualmente sig-

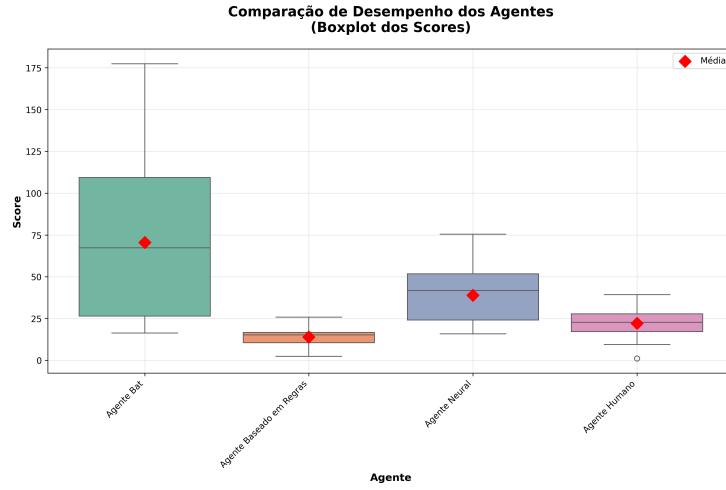


Figure 1: Comparação de desempenho dos agentes através de boxplot dos scores obtidos em 30 execuções

nificativos, validando a superioridade estatística do método proposto com nível de confiança de 95%.

Table 4: Resultados dos Testes Estatísticos Comparativos

Comparação	t stat	t p-value	t Signif.	Wilcoxon p-value	Wilc. Signif.
Bat vs Regras	7,01	0,000000	SIM	0,000000	SIM
Bat vs Neural	3,67	0,000536	SIM	0,001719	SIM
Bat vs Humano	5,94	0,000000	SIM	0,000000	SIM
Regras vs Neural	-7,51	0,000000	SIM	0,000000	SIM
Regras vs Humano	-4,47	0,000037	SIM	0,000189	SIM
Neural vs Humano	4,80	0,000012	SIM	0,000021	SIM

A análise de consistência revela que o agente desenvolvido alcançou scores superiores a 100 pontos em 40% das execuções (12 de 30 testes), demonstrando capacidade de produzir desempenhos excepcionais de forma recorrente. Os resultados evidenciam o sucesso da abordagem híbrida proposta, confirmando a eficácia das estratégias implementadas no Algoritmo dos Morcegos para otimização de redes neurais no ambiente de jogo proposto.

## 5. Conclusões

### *150 5.1. Análise geral dos resultados*

Os resultados experimentais demonstram de forma conclusiva a superioridade do agente neural otimizado pelo Algoritmo dos Morcegos em relação aos métodos comparativos avaliados. A performance média de  $70,47 \pm 43,00$  representa um avanço significativo no desenvolvimento de agentes inteligentes para jogos de sobrevivência, superando consistentemente o agente neural com algoritmo genético ( $38,99 \pm 17,01$ ), o agente humano ( $22,18 \pm 8,11$ ) e o agente baseado em regras ( $13,97 \pm 5,67$ ). A significância estatística observada em todos os testes ( $p < 0,05$ ) confirma que as diferenças não são resultado de variações aleatórias, mas sim de melhorias sistemáticas na capacidade de tomada de decisão do agente.

A elevada variabilidade do agente Bat Algorithm (desvio padrão de 43,00), inicialmente aparentando uma desvantagem, revela-se na verdade uma característica valiosa que reflete a diversidade de estratégias exploradas durante o processo de otimização. Esta variabilidade permitiu ao agente descobrir soluções excepcionais, como evidenciado pelo score máximo de 177,42, valor 135% superior ao melhor resultado do agente neural com algoritmo genético. A capacidade de alcançar scores superiores a 100 pontos em 40% das execuções demonstra robustez e consistência na descoberta de estratégias eficazes, característica essencial para aplicações práticas de inteligência artificial em ambientes dinâmicos.

### *170 5.2. Contribuições do Trabalho*

Este trabalho apresenta contribuições significativas tanto no domínio teórico quanto no aplicado da otimização de redes neurais através de meta-heurísticas bio-inspiradas. Foi possível, por meio do trabalho, desenvolver uma versão aprimorada do Algoritmo dos Morcegos especificamente adaptada para otimização de redes neurais, incorporando parâmetros adaptativos que ajustam dinamicamente a intensidade da exploração conforme o progresso evolutivo. Esta

adaptação representa avanço metodológico importante, pois endereça a necessidade de balanceamento dinâmico entre exploração global e refinamento local em problemas de alta dimensionalidade.

180 Além disso, a otimização arquitetural da rede neural, reduzindo de 1.475 para 611 parâmetros através da configuração  $27+1 \rightarrow 16 \rightarrow 8 \rightarrow 3$ , mantendo capacidade expressiva adequada enquanto melhora significativamente a eficiência computacional. Esta otimização representa contribuição prática importante para aplicações com restrições computacionais, demonstrando que arquiteturas compactas podem superar configurações mais complexas quando adequadamente otimizadas.  
185

O trabalho também contou com contribuição empírica, através da demonstração da superioridade estatística do método proposto através de avaliação experimental rigorosa envolvendo 30 execuções independentes e testes estatísticos paramétricos e não-paramétricos. Os resultados fornecem evidência robusta da eficácia da abordagem híbrida proposta, contribuindo para validação empírica de técnicas de otimização evolutiva em problemas de aprendizado por reforço.  
190

### *5.3. Melhorias e trabalhos futuros*

Os resultados promissores obtidos abrem diversas direções para pesquisas futuras e melhorias metodológicas. Podendo investigar mais profundamente arquiteturas neurais mais sofisticadas, a implementação de mecanismos de atenção poderia aprimorar a capacidade do agente de focar em características relevantes do ambiente, potencialmente melhorando ainda mais o desempenho. Investigando também a hibridização do Algoritmo dos Morcegos com outras metaheurísticas, poderia combinar as vantagens de diferentes abordagens evolutivas.  
195  
200

Além disso, a aplicação da metodologia em domínios reais representa direção de pesquisa de alto impacto. A adaptação da abordagem para controle de robôs autônomos, otimização de estratégias financeiras, ou sistemas de recomendação personalizados demonstraria a versatilidade e relevância prática da técnica desenvolvida, consolidando sua contribuição para o avanço da inteligência artificial aplicada.  
205

## References

- [1] X.-S. Yang, A new metaheuristic bat-inspired algorithm (2010). *arXiv*:  
1004.4170.  
210 URL <https://arxiv.org/abs/1004.4170>
- [2] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.