Relatório: Algoritmo de Aprendizado por Reforço Multi-Agente Speaker-Listener

Luis Sante & Joel Perca & Andres de la Puente

29 de novembro de 2025

1 Introdução e Objetivo

O objetivo deste projeto é desenvolver e implementar um novo algoritmo de Aprendizado por Reforço Multi-Agente (MARL) ou otimizar um algoritmo existente, como o **MATD3 (Multi-Agent Twin Delayed DDPG)**, para o ambiente **Speaker-Listener**. O critério de sucesso é que o agente *Listener* consiga navegar até o alvo de forma mais eficiente, ou seja, alcançar uma **pontuação média populacional superior a -60**.

2 Configurações e Metodologia de Treinamento

O treinamento foi conduzido em múltiplos ciclos, cada um com uma combinação distinta de hiperparâmetros. Cada iteração representa milhares de passos no ambiente, totalizando **milhões de interações**. O objetivo central foi avaliar a **convergência** e a **estabilidade** do agente *Listener* ao longo de diferentes regimes de treinamento.

2.1 Variações nas Configurações ao Longo do Treinamento

As execuções apresentadas neste relatório utilizaram ajustes iterativos nos principais componentes do algoritmo. Em cada rodada foram modificados elementos estruturais (como a arquitetura da rede), dinâmicos (exploração e taxas de aprendizado) e evolutivos (tamanho da população e frequência de atualização). O foco não foi comparar configurações específicas, mas **examinar como diferentes regimes afetam o comportamento do sistema Speaker-Listener**.

Categoria	Ajuste Realizado	Razão
Arquitetura da Rede	Capacidade e profundidade	Testar maior expressividade
População Evolutiva	Nº de agentes	Aumentar diversidade para HPO
Batch Size	Tamanho do batch	Estabilidade do gradiente
Exploração	Intensidade do ruído	Exploração vs. convergência
Learning Rates	Atores e críticos	Velocidade de aprendizado
TAU (Redes-Alvo)	Ritmo de atualização	Suavizar adaptação
Policy/Update Freq	Freq. de atualização	Estabilidade da política
Evo Steps	Freq. de ciclos evolutivos	Controle da busca evolutiva

Padrões observados ao longo dos experimentos:

- **Estabilidade crescente após 50 iterações**.
- **Maior volatilidade** com ruído de exploração elevado ou menor atraso do TD3.
- **Recuperação rápida** após quedas profundas.

Os treinamentos de 0–2M, 0–3M e 0–5M iterações são **variações controladas** dentro do mesmo experimento.

3 Análise de Resultados

Os resultados são apresentados em gráficos que mostram a evolução das pontuações médias ao longo do treinamento, em diferentes escalas.

3.1 Treinamento Inicial (0–2 milhões de iterações)

O desempenho melhora rapidamente nas primeiras iterações: de aproximadamente **-100** para a faixa entre **-50 e -75**.

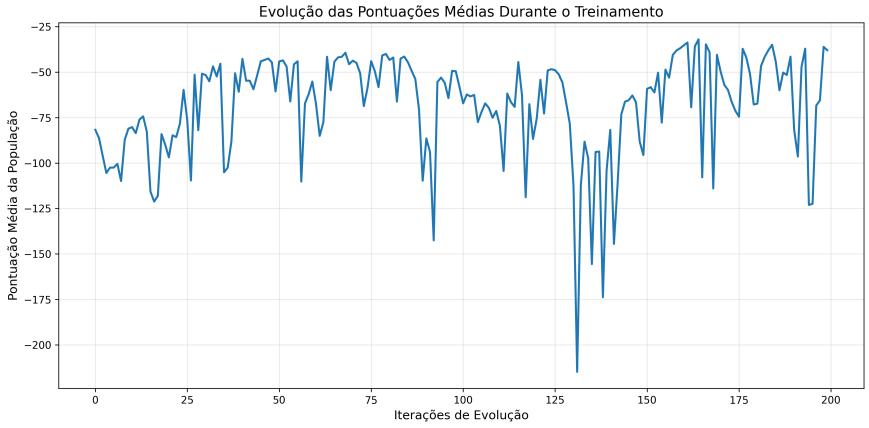

Figura 1: Evolução das Pontuações (0–2M)

Observa-se alta volatilidade, com quedas até **-210**. Apesar disso, a meta de **-60** é atingida já por volta da iteração 20.

3.2 Treinamento Ampliado (0–3 milhões de iterações)

Após 50 iterações, a média estabiliza entre **-50 e -75**. Quedas severas abaixo de **-250** indicam falhas críticas ocasionais, mas o sistema recupera rapidamente.

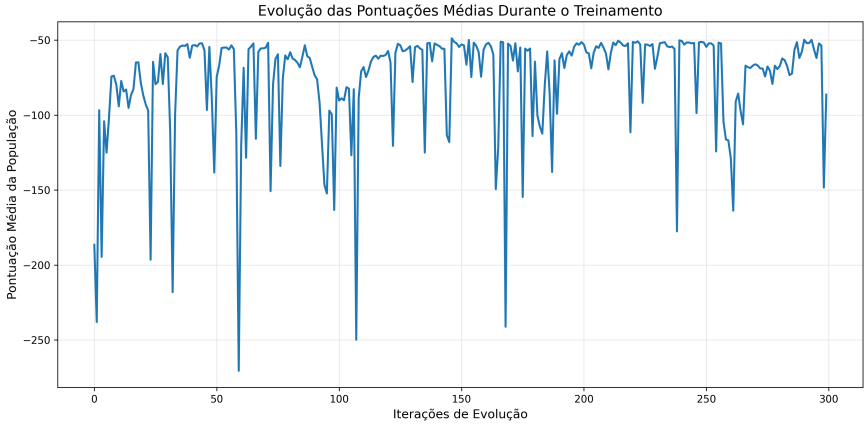

Figura 2: Evolução das Pontuações (0–3M)

3.3 Treinamento Estendido (0–5 milhões de iterações)

O sistema mantém pontuações entre **-30 e -70**. O limiar de **-60** é superado de modo consistente ao longo do treinamento.

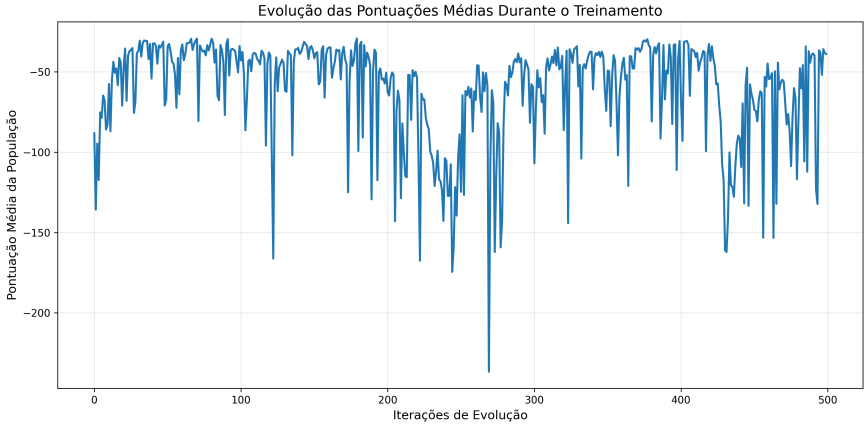

Figura 3: Evolução das Pontuações (0–5M)

4 Conclusão

O algoritmo demonstrou desempenho robusto no ambiente Speaker-Listener. A meta de manter a pontuação acima de **-60** foi cumprida de forma consistente ao longo de 5 milhões de iterações.

5 Trabalhos Futuros

1. Reduzir a volatilidade extrema (quedas até -275).
2. Aplicar médias móveis para curvas mais suaves.