

# Aplicação de inteligência artificial em jogos de estratégia complexos

Guilherme Vilar Balduino<sup>1</sup>, Roberto Yamamoto<sup>1</sup>

<sup>1</sup>Departamento de Computação – Universidade Federal de São Carlos(UFSCar)

**Abstract.** *This article's objective is explain what is necessary for an artificial intelligence (AI) be able to react in complex situations like strategy games. The article will focus in the game Starcraft, a complex real-time strategy game (RTS) where the player must be able to control numerous units that perform different functions, to seek the best way to collect materials and develop a tactic to destroy all the enemies structures. For an AI to deal with these situations, it applies Reinforcement Learning and, because of this, it could defeat professional Starcraft players.*

**Resumo.** *Esse artigo tem como objetivo explicar de maneira geral o que é necessário para uma inteligência artificial (IA) ser capaz de atuar em situações complexas como jogos de estratégia. O foco do artigo será o jogo Starcraft, um jogo de estratégia em tempo real (RTS) complexo em que o jogador deve ser capaz de controlar inúmeras unidades que realizam funções distintas, buscar o caminho mais eficiente para coletar materiais e desenvolver uma tática para destruir as estruturas dos inimigos. Para que a inteligência artificial seja capaz de lidar com essas situações, ela aplica a aprendizagem por reforço e, usando esse aprendizado, foi capaz de vencer jogadores profissionais de Starcraft.*

## 1. Contextualização e projeto

Em janeiro de 2019, a inteligência artificial da empresa DeepMind, AlphaStar, derrotou jogadores profissionais de Starcraft em uma série de onze partidas[Wang 2019]. Uma IA já superou anteriormente humanos profissionais em outros jogos de estratégia como xadrez, que é feito em sistema de turnos; mas o caso de Starcraft é diferente por ser um jogo de estratégia em tempo real[Santos 2017]. A grande dificuldade de criar uma IA para um jogo de estratégia em tempo real é o grande número de fatores que influenciam na dinâmica do jogo.

No caso de Starcraft, a dinâmica do jogo se altera de acordo com as diferentes estratégias para cada raça presente no jogo(Terran, Zerg e Protoss), conhecimento parcial do cenário, tal que o jogador somente tem conhecimento das regiões próximas aos seus edifícios e unidades, e os diversos tipos de unidades, tais quais tem diferentes utilidades, como os coletores(VCEs, Zangões e Sondas) e unidades aéreas(Viking, Corruptor, Fênix) e com essa gama enorme de opções é preciso uma IA que consiga arquitetar uma estratégia eficiente e que possa prever a melhor sequência de ações para alcançar o objetivo, ou seja destruir todos os edifícios de todos os adversários.

A ideia principal do projeto é explicar como a IA é capaz de resolver as situações variadas de jogo usando aprendizado por reforço[Roldán Montaner 2018] e que é possível aplicar outros tipos de aprendizado para solucionar o problema, como o aprendizado de máquina.

## 2. Aprendizado por reforço

Esse tipo de aprendizado de máquina foi criado com base em como os humanos e os animais aprendem a tomar decisões, mediante a ganhos positivos e negativos.

Em um determinado momento, o agente se encontra em um estado e mediante a isso decide realizar uma ação dentre todas as disponíveis no momento. Em resposta o ambiente retorna uma recompensa imediatamente. Assim o agente vai aprendendo a escolher as ações que maximizam a sua soma de recompensas.

Em suma, o seu objetivo é encontrar uma política para escolher ações que tragam ao agente a recompensa máxima a longo prazo.

O método de aprendizado é dividido em quatro elementos principais[Oliveira et al. 2018] que definem a decisão que será tomada pela inteligência artificial.

### 2.1. Política

Define como o agente deve se comportar mediante a uma determinada situação. Representam as regras do ambiente.

### 2.2. Função recompensa

Define o objetivo a ser atingido no momento atual, geralmente aplicada para tomar decisões em curto prazo.

### 2.3. Função valor

Função responsável por definir a recompensa que o agente pode acumular a partir de um estado, desse modo é capaz de prever em que estado a recompensa será maior. A função é utilizada para tomar decisões a longo prazo.

### 2.4. Modelo do ambiente

Uma representação do ambiente no qual o agente foi inserido e a lista de ações que o agente é capaz de realizar.

## 3. Definições matemáticas

O aprendizado por reforço tem um forte embasamento matemático[Oliveira et al. 2018], o qual será analisado agora.

Inicialmente é necessário avaliar o valor das ações para escolher qual deve ser realizada em um momento específico com o intuito de calcular o quão boa é a ação. Isso é determinado com o cálculo a seguir:

$$Q_t(a) = \frac{R_1 + R_2 + \dots + R_{K_a}}{K_a}$$

Figure 1. Média das recompensas geradas

$R_1 + R_2 + \dots + R_K$  é a soma das recompensas obtidas com a ação e  $K$  representa a quantidade de vezes que a ação foi escolhida, se a ação nunca foi escolhida então  $Q_t$  receberá um valor pré-estabelecido.

Com essa função definida, são determinados métodos para escolher as ações que serão tomadas em algum momento do problema. As abordagens mais comuns são a Greedy (escolhe a ação com maior recompensa), e-greed (escolha baseada no maior aproveitamento, com probabilidade  $1 - \epsilon$ , sendo  $\epsilon$  a probabilidade de escolher uma ação aleatória) e softmax (ainda acontece a ação gulosa, mas outras ações são escolhidas baseadas nas suas respectivas estimativas de distribuição).

O agente tem o objetivo de maximizar a recompensa a longo prazo, de modo que é possível definir uma função que descreva o retorno esperado:

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \dots + R_T$$

**Figure 2. Retorno esperado após T ações**

$T$  é a última ação a ser tomada para solucionar o problema e  $G_t$  é a soma das recompensas recebidas depois do instante  $t$ . Contudo também é necessário introduzir o conceito de desconto, essa taxa (representada por gama) é um valor entre 0 e 1, e determina o valor atual de futuras recompensas, assim fazendo o agente tomar decisões a curto ou a longo prazo, o que modifica a função anterior:

$$G_t = R_{t+1} + \gamma^2 R_{t+2} + \gamma^3 R_{t+3} + \dots + \gamma^n R_T = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

**Figure 3. Retorno esperado incluindo descontos**

#### 4. Aplicação em Starcraft

Para o jogo Starcraft, é necessário que a inteligência artificial tenha um conhecimento sobre todas as unidades presentes no jogo. Sobre cada unidade, ela deve saber qual a sua função e o que o agente pode fazer com essa unidade[Koeman et al. 2018].

Com o conhecimento sobre as unidades presentes em Starcraft, a IA pode buscar a solução mais eficiente a longo prazo baseada no estado atual em que o programa se encontra. Mas para que o agente possa realizar os movimentos necessários para a solução, é preciso que ele controle inúmeras unidades simultaneamente. Para isso a inteligência artificial deve possuir uma alta performance.

Outro fator que dificulta a análise da IA é a constante variação de situações que uma partida de Starcraft oferece. Durante a partida um jogador precisa pensar no momento certo para fortalecer suas defesas, coletar materiais, criar outras unidades, entre outras funções que o jogo permite. Por esse motivo não é vantajoso à inteligência artificial pensar em soluções a curto prazo e, com isso, um sistema de aprendizado comum não encontraria as soluções mais eficientes para que uma IA pudesse vencer uma partida.

A aplicação de aprendizado por reforço foi essencial para o desenvolvimento da inteligência artificial. Como esse tipo de aprendizado foca em melhores resultados a longo

prazo, é utilizado para que a IA consiga realizar as funções necessárias para se manter durante a partida enquanto guarda recursos caso seja necessário mudar de estratégia. A mudança de estratégia é muito comum em jogos de estratégia em tempo real, em que o jogador muda de estado constantemente.

Para desenvolver a inteligência artificial AlphaStar, inicialmente foi um aprendizado supervisionado em que a IA analisava partidas de Starcraft entre pessoas, em seguida ela competiu contra outros agentes gerados por meio de observações de partidas e, com esse conhecimento, foi capaz de desenvolver novas estratégias a partir das estratégias usadas pelos agentes[AlphaStar team 2019]. Dessa forma, o agente pode escolher o movimento mais eficiente a longo prazo baseado no ambiente em que se encontra.

## **5. Conclusão**

A evolução da tecnologia permitiu que métodos e cálculos mais avançados fossem aplicados em inteligências artificiais. A vitória da AlphaStar representa os resultados dessa evolução.

No entanto, a aplicação da aprendizagem por reforço vai muito além de vencer jogos. Ela pode ser útil para solucionar problemas reais, como encontrar uma forma de economizar energia em uma indústria, por exemplo.

O aprendizado da máquina também pode ajudar a desenvolver os humanos, pois a inteligência artificial é capaz de chegar a soluções que um humano não chegaria. Em Starcraft existem inúmeras estratégias possíveis e os jogadores profissionais abordam apenas algumas delas em competições, porém tais estratégias foram derrotadas pelas táticas usadas pela AlphaStar, o que significa que a máquina foi capaz de chegar em uma forma diferente de derrotar o inimigo que os jogadores profissionais de Starcraft não esperavam.

A inteligência artificial é uma área ainda em desenvolvimento e, por isso, tem muito a evoluir. Atualmente é possível ver máquinas jogando jogos de estratégia a nível profissional, mas no futuro outras empresas criarão novas IA que vão superar jogadores profissionais em outros jogos competitivos.

## **References**

- AlphaStar team, T. (2019). Alphastar: Mastering the real-time strategy game starcraft ii. <https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/>.
- Koeman, V. J. et al. (2018). Starcraft as a testbed for engineering complex distributed systems using cognitive agent technology.
- Oliveira, G. P. d. et al. (2018). Métodos de inteligência artificial aplicados em jogos baseados em turnos.
- Roldán Montaner, C. (2018). Deep reinforcement learning ia para starcraft 2. B.S. thesis, Universitat Politècnica de Catalunya.
- Santos, V. H. d. (2017). Experimentos com aprendizado por reforço em cenário de combate de starcraft. B.S. thesis, Universidade Federal do Rio Grande do Norte.
- Wang, B. (2019). Deep mind artificial intelligence wins 10 out 11 games versus pro human players. <https://www.nextbigfuture.com/2019/01/deep-mind-artificial-intelligence-wins-10-out-11-games-versus-best-human-players.html>.