

Tema:

**Aplicação de *Machine Learning* em jogos do gênero AutoChess**



Figura 1. Interface de configuração de cenários e visualização do treinamento.

## Motivação

Em jogos digitais de forma geral, é comum encontrar personagens que são controlados por um algoritmo programado pelos desenvolvedores que apresenta um conjunto de ações padrão e bem definido. Entretanto, quando o jogador depende deste comportamento pré-definido para alcançar o objetivo dentro do jogo, sem poder fazer alterações, isto pode ser um problema pois não permite adequá-lo ao seu estilo de jogo. Isso é bem evidente em jogos do gênero AutoChess em que após posicionar as peças no tabuleiro, elas seguem um comportamento padrão e previsível.

## Objetivo

O objetivo deste trabalho é implementar uma versão simplificada do jogo de gênero AutoChess com um sistema de treinamento das peças que aplique técnicas de *Machine Learning* (ML) e permita que o usuário apresente o comportamento desejado para cada uma delas de forma fácil que não necessite de experiência prévia de uso do sistema.

**Integrantes:** Elton Yoshio Okawa  
Hélio Jin Wu Kim

**Professor Orientador:** Prof. Dr. Ricardo Nakamura

## Método

A interface de treinamento foi desenvolvida com a engine de jogos UnrealEngine™, foram testadas diferentes modelagens para o problema e o treinamento em si utiliza algoritmos de *Inverse Reinforcement Learning* (IRL) e *Reinforcement Learning* (RL) chamados *Apprenticeship Learning* e *Sarsa Lambda*, respectivamente.

Para criar um novo comportamento, o jogador deve demonstrar trajetórias que alimentam o algoritmo de ML que quando converge, resulta em um comportamento generalizado que as representa, como mostra a Figura 2.

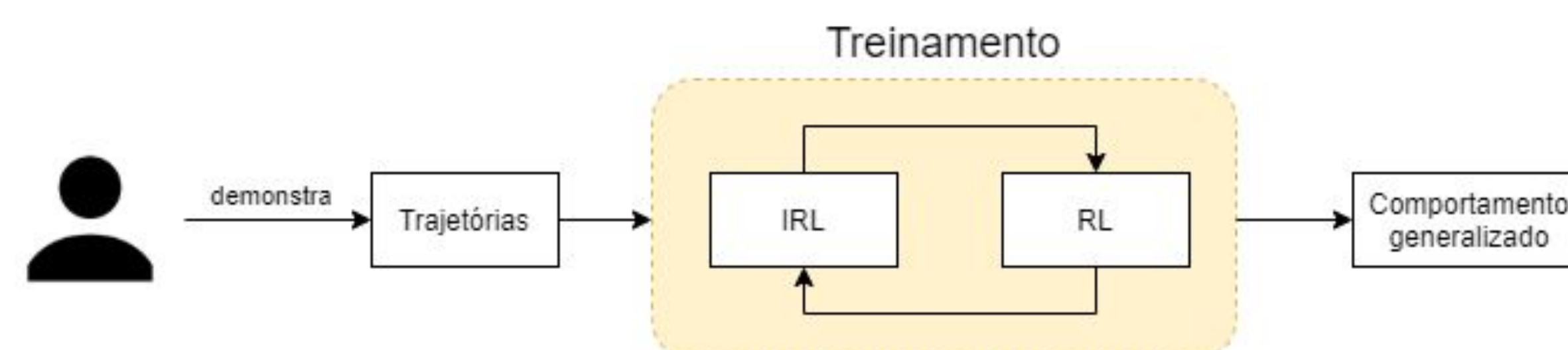


Figura 2. Fluxo de treinamento.

## Interface de Treinamento

Foi desenvolvida uma interface para configuração e visualização do treinamento em que o jogador tem à disposição dois modos importantes, um em que ele toma o controle do personagem e demonstra o comportamento desejado e outro, como mostra a Figura 1, que se cria cenários de treinamento e converge o algoritmo de ML.

## Resultados

Com a modelagem do problema atual, o agente é capaz de replicar comportamentos e posicionamentos mais simples demonstrados pelo expert, porém apresenta ações inesperadas em algumas situações.