SimToReal: Transferring Learning to play robot soccer

PROPOSTA DE TRABALHO DE GRADUAÇÃO

Aluno: José Nilton de Oliveira Lima Júnior (jnolj@cin.ufpe.br)
Orientador: Hansenclever de França Bassani (hfb@cin.ufpe.br)
Co-orientadora: Edna Natividade da Silva Barros (ensb@cin.ufpe.br)
Área: Aprendizagem por Reforço / Robótica

28 de março de 2019
Abstract

Recent advances in reinforcement learning (RL) are responsible for achieving good results in many game benchmarks, such as Chess, Go, Starcraft (even at the professional human level). However, applying RL in an environment with real-world actuators - like robots - comes with a new layer of complexity. Training in real-world is unfeasible due to equipment degradation and time of training required by RL algorithms. However, intrinsic discrepancies between the simulated and real-world environments prevent a direct transfer of a policy learned in a simulated environment to robots - the so-called Sim-To-Real problem. This project aims at evaluating the Sim-To-Real problem in the robot soccer domain. The objective is to transfer the policy learned with simulated robots to a real robot, aiming to obtain a transferred policy that is able to take the ball to the enemy's goal, as efficiently and effectively as possible.
Resumo

Avanços recentes em aprendizado por reforço (AR) foram responsáveis por alcançar grandes resultados em vários marcos de inteligência artificial em jogos, tais como Xadrez, Go e Starcraft. Porém, aplicar AR em um ambiente com atuadores do mundo real - como robôs - traz uma nova camada de complexidade. O treinamento no ambiente real é frequentemente inviável devido a degradação dos robôs e tempo de treinamento necessários por algoritmos de reforço. Ainda, diferenças intrínsecas entre o ambiente real e o simulado impedem a transferência de uma política aprendida num ambiente simulado para robôs - este problema é chamado de Sim-To-Real. Esse projeto almeja avaliar o problema Sim-To-Real no contexto de futebol de robôs. O objetivo é transferir uma política treinada com robôs simulados para um robô real, visando obter uma política transferida que é capaz de levar a bola para o gol inimigo.
Introduction

Reinforcement learning is the subfield of machine learning that studies the problem faced by an agent that must learn a behavior - also called policy - through trial-and-error while interacting with its environment [1]. To enable learning, the environment gives to the agent information about its current state and the agent interacts with the environment by taking actions over time. The environment also produces a reward signal, giving positive rewards to good actions and penalties to the bad ones. The learner is not told which actions to take, but instead it must discover which actions yield more positive rewards by trying them - this is why it is called reinforcement learning. Therefore, the agent's goal is to learn how to maximize the cumulative reward over time [2].

Recent advances in reinforcement learning are responsible for achieving impressive results in many benchmarks in artificial intelligence and even beating professional players of Chess [3], Go [4] and Starcraft [5]. Recent works turn focus on achieving agents capable to coordinate with each other in a multi-agent system. Research in games like Dota [6] and robot soccer simulation [7] presents the most impressive results in this subfield.

However, when it comes to applying reinforcement learning in an environment with real-world actuators - like robots - a new layer of complexity is placed. Reinforcement learning algorithms are expensive in terms of power consumption and time taken for training. Also, it is dangerous to robot integrity as it performs many random actions in their initial training phase. Thus, there are no reports of successful end-to-end reinforcement learning system able to play games in real-world, such as robot soccer.

To overcome the problems of training reinforcement algorithms in real-world, the idea of using a simulated environment comes in mind. Unfortunately, discrepancies between physics simulators and the real-world make the behavior learned in simulation - also called policy - inadequate for the real-world. Even with strong system identification, the real-world has unmodeled physical effects like non-rigidity, gear backlash, wear-and-tear and fluid dynamics that are not captured by current physics simulators [8]. Moreover, real-world observations rely on sensory data, which is usually noisy. This discrepancy between the simulated and real world is called the reality gap.

The problem of trying to minimize the reality gap is called Sim-To-Real. The state-of-art in this area shows several strategies to close the gap. One possible approach is to use evolutionary algorithms to evolve the agent behavior, the simulation, or both [9], [10]. On the other hand, some research focuses on Domain randomization, which aims at developing an agent that learns to perform well on different environments by
training it in several variations of the simulated environment [8], [11], [12]. There are also reinforcement learning algorithms developed specially to learn in real-world environments, reducing the number of random actions to prevent robot damage [13]. Each approach usually trades between training time and solution quality, not having a perfect solution. Therefore, it is necessary to evaluate which approach is more suitable for each problem domain.
Objectives

This project aims at evaluating the Sim-To-Real problem in the robot soccer domain, where high competitiveness and cooperation require precise actions. The objective is to transfer the policy learned by a neural network trained with simulated robots to a real robot, aiming to obtain a transferred policy that is able to take the ball to the enemy’s goal, as efficiently and effectively as possible. As there are several variations of soccer robots, this research is based on the IEEE - Very Small Size Soccer (VSSS).

The VSSS is defined as a soccer game in a field with dimensions 170cm x 130cm, where teams of three robots should lead an orange ball to the enemy goal without human intervention. The maximum size of each robot is 7.5x7.5x7.5cm. In order to allow the robots to play autonomously, a camera is positioned above the field to acquire images of the game and send them to a computer, which is the only component allowed to send commands to the robots. The Fig.1 illustrates the setup for a VSSS game.

Fig.1 - VSSS game setup example Fonte: [17]
Methodology

This project will be developed according to the following steps:

1. **Literature Review** - This step aims at understanding the state-of-art research in Sim-To-Real field by pointing out the main approaches and evaluating their pros and cons.

2. **Domain analysis** - The objective of this step is to evaluate quantitatively the discrepancy between the simulation and real environment. The command rate, robot physics, actuators response, and robot speeds will be evaluated in both domains (simulation and real). After that, a comparison of the obtained results will be made to choose the best approach to perform the transfer.

3. **Transfer implementation** - This step aims at implementing the proposed Sim-To-Real transfer method to transfer a policy trained in the simulation to the real robots. The method will be chosen accordingly to the domain analysis result.

4. **Model evaluation** - This step aims at evaluating the real policy performance in comparison to the simulated one. The mean agent reward, mean time to bring the ball to the goal, mean of goals performed in 5 minutes, and mean robot speed will be used as metrics.

5. **Model tuning** - This step aims at optimizing the model according to the performed experiments results, using domain knowledge to achieve better results.

6. **Dissertation writing** - In this step, a dissertation in is written following the Computer Engineering dissertation standards.

To perform the experiments, the VSS-Simulator [14] will be used. To the real-world environment, the infrastructure of the VSS game will be provided by the RoboCLn team. The robot version will be equal to the Latin America Robots Competition (LARC) 2018 RoboCLn robot, described in [15]. To perform reinforcement learning, two Gym Environments will be developed[16] in order to standardize the code and reduce the software difference. The policy to be transferred to the robot will be provided by the professor Hancenlever Bassani, which comes from partial results of another research that uses the same simulator. Therefore, the training of the policy is not part of this work.
## Schedule

<table>
<thead>
<tr>
<th>Activities</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
<th>July</th>
</tr>
</thead>
<tbody>
<tr>
<td>Literature Review</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Domain analysis</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Transfer implementation</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Model evaluation</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Model tuning</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Dissertation Writing</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Presentation Preparation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
References


Possível Avaliador

Prof. Tsang Ing Ren