Relatório - Descrição

Valdinei Freire

4 de Março de 2016

1 Processos Markovianos de Decisão e Aprendizado por Reforço

Processos Markovianos de Decisão (Markov Decision Process - MDP) considera problemas em ambientes: *completamente observável*, único agente, **conhecido**, **estocástico**, *discreto*, sequencial, estático.

Um MDP é modelado por:

- um conjunto de estados $\mathcal{S}$
- um conjunto de ações $\mathcal{A}$
- uma função de transição $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$
- uma função recompensa $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

Eventualmente pode-se considerar ainda:

- um estado inicial $s_0 \in \mathcal{S}$ ou $\beta(s) = \Pr(s_0 = s)$
- um conjunto de estados metas $\mathcal{S}_G \in \mathcal{S}$

Aprendizado por Reforço (Reinforcement Learning - RL) considera problemas em ambientes: *completamente observável*, único agente, **desconhecido**, **estocástico**, *discreto*, sequencial/episódico, estático.

Um problema de RL é modelado por um MDP, no qual a função de transição e a função recompensa são desconhecidas.

2 Ambientes

Para a execução desse trabalho considera-se dois ambientes: navegação robótica e futebol de robô. Ambos em um ambiente simulado e discreto.

2.1 Navegação Robótica

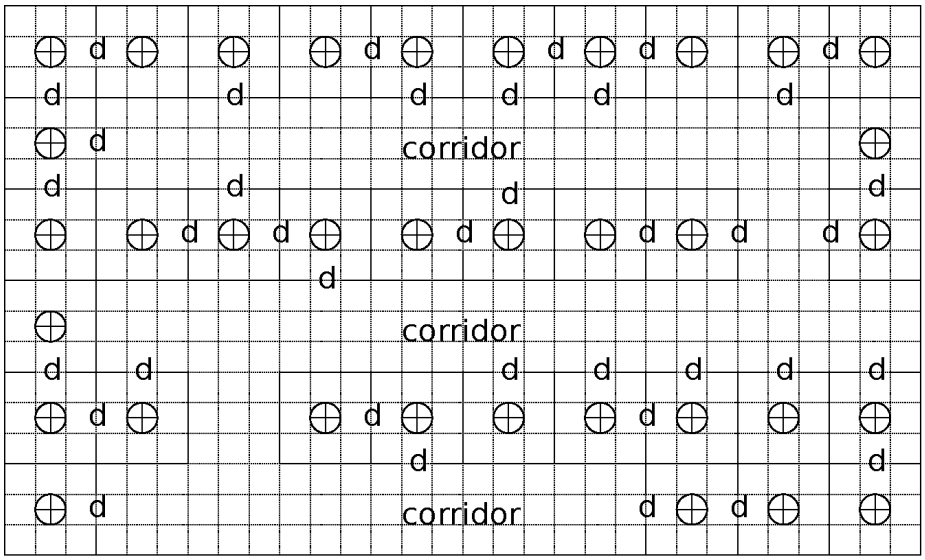
Ambientes com várias salas e um corredor, no qual o agente deve encontrar uma **política** para chegar no centro de cada sala.

Os estados são enumerados.

As ações são:

1. vá para um espaço vazio na direção oposta a meta
2. vá para uma porta na direção oposta a meta
3. vá para uma sala na direção oposta a meta
4. vá para um corredor na direção oposta a meta
5. vá para um espaço vazio na direção da meta
6. vá para uma porta na direção da meta
7. vá para uma sala na direção da meta
8. vá para um corredor na direção da meta

As ações obtém sucesso com probabilidade 0,9.

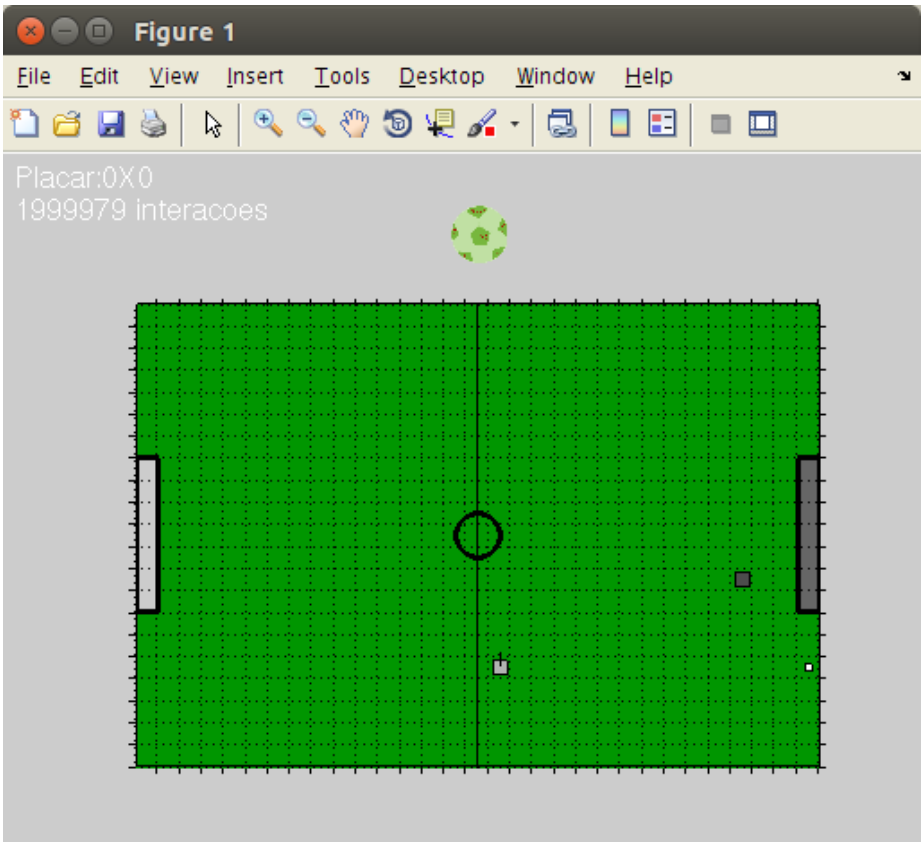


Material: arquivos para criar um modelo para o MDP.

2.2 Futebol de Robô

Ambiente que simula um jogo de futebol em um ambiente discretizado.

Pode-se escolher o tamanho do campo e quantidade de jogadores em cada lado.



Existe um agente baseado em regras já implementado.

Os estados são fatorados: posição (x,y) de cada jogador, posição (x,y) da bola, velocidade da bola e direção da bola.

As ações:

1. norte
2. sul
3. leste
4. oeste
5. chuta norte
6. chuta sul
7. chuta leste
8. chuta oeste

9. chuta nordeste
10. chuta sudeste
11. chuta noroeste
12. chuta sudoeste
13. rouba a bola

Todas as ações possuem resultados estocástico.

- vai para direção escolhida com probabilidade 0,95
- jogador carrega bola com probabilidade 0,9
- jogador acerta chute com probabilidade 0,9
- jogador toma a bola com probabilidade 0,5
- bola tem velocidade dependendo do campo e diminui com o tempo
- bola avança com probabilidade 0,5

Material: Arquivos para simular o jogo de futebol.

3 PARTE I - Processos Markovianos de Decisão

3.1 Navegação Robótica

Implementar os algoritmos: Value Iteration e Policy Iteration.

Testar e avaliar os algoritmos nos dois ambientes fornecidos no trabalho. A avaliação deve considerar a quantidade de iterações até a convergência para diferentes valores de ϵ.

3.2 Futebol de Robô

Considere um jogo de futebol que acaba assim que o seu time faça gol ou em uma quantidade finita de tempo (horizonte finito).

Utilizando um agente baseado em regras (criado pelo grupo), implemente alguma técnica baseada em *Monte Carlo Tree Search* (MCTS) para obter uma política parcial ao considerar um estado inicial (seu jogador no centro do campo com a posse da bola e o jogador adversário no centro do seu lado do campo). Utilize o agente baseado em regra para executar os *rollouts*.

Testar e avaliar os algoritmos nos três ambientes fornecidos no trabalho, comparando o agente baseado em regra, com a versão baseada em planejamento.

4 PARTE II - Aprendizado por Reforço

4.1 Navegação Robótica

Implementar os algoritmos: Q-Learning e Sarsa(λ).

Testar e avaliar os algoritmos nos dois ambientes fornecidos no trabalho. Teste diferentes estratégias de exploração-explotação e diferentes estratégias para taxa de aprendizado.

4.2 Futebol de Robô

Implementar o algoritmo: Q-Learning, e testá-lo no ambiente mais simples: 7×11.

A partir dos fatores canônicos do ambiente, criar novos fatores que sejam relevantes para tomada de decisão (o agente baseado em regras pode ajudar).

Implementar o algoritmos: Sarsa(0) com aproximação de função.

5 Relatório

Um relatório resolvendo os problemas acima deve ser escrito com no máximo OITO páginas (em formato disponível no TIDIA) e submetido no TIDIA. O relatório deve estar no formato PDF e deve descrever como os resultados foram obtidos. Também deve ser submetido os código fontes utilizados para produzir os resultados.

Cada grupo pode ser formado por no máximo 4 pessoas.