Machine Learning Controllers for Autonomous Surface Vessels

Haakon Robinson

Preprint submitted to Elsevier *October 6, 2018*

Contents

1. Introduction

Study the possibilities of applying reinforcement learning to controls problems involving marine surface vessels.

2. Background

2.1. Modeling Marine Surface Vessels

2.2. Deep Learning and Decision Making

2.3. Reinforcement Learning

2.3.1. Definitions

Agent Environment State Action Reward

2.3.2. Markov Processes

A Markov Process is a model that describes the possibly states of the environment, and the probability of transitioning from one state to another. The state is typically a vector of numbers. The transitions are stochastic, such that at every time step there is a probability of transitioning to another state. The transition can also map a state onto itself.

The state must satisfy the **Markov Property**, also known as the memoryless property. This means that the information contained in the state must be independent of past states. In other words, the state must "summarise" the situation, telling you everything you need to know.

The discrete case? Continuous case?

2.3.3. Markov Decision Processes

A Markov Decision Process (or MDP) is a mathematical tool that can be used to model decision making. It extends the notion of Markov Chains by adding the notions of choice and reward.

2.3.4. The Bellman Equation

Equation that represents the value of being in a particular state, based on the reward given for being in that state and the value of the best possible adjacent state that can be reached by taking one action, usually multiplied by a discount factor. This results in a recursive equation, as the value of the state depends on the value of its surrounding states. Typically, one can start from the goal state and propagate the state values "backwards". .

2.3.5. Q Learning

Replace the state value function in the Bellman equation with the Q function. This can be interpreted as "action quality".

2.3.6. Q learning with neural networks

Instead of using a table to store the policy, the policy is "stored" in a neural network. This has consequences for stability.

3. Literature Review

3.1. Deep Reinforcement Learning

(Human level control through deep reinforcement learning)

3.2. Set-Based Tasks for Controls

Framework for implementing several modes of operation based on the current active tasks. Allows for the use of several different control systems at once, each one specialising in different combinations of tasks. This is a good way to combine the strengths of rule-based and ML-based controllers.

4. Problem Formulation

In a navigation problem, one generally has a destination that you wish to get your system to. If the environment is deterministic, using some kind of path planning method like A* works well, as long as the "obstacles" are known. When the environment is dynamic and unknown, this is not as easy. The "global" path may still be there, but there might be many small local obstacles that need to be avoided. A local avoidance system is therefore needed.

Potential fields might lead to oscillatory behavior.

Dynamic window assume no sideways velocity, which is difficult in the context of ocean currents. Also computationally heavy, but works well with COLREGS.

5. Simulation and Testing Environment

6. Deep Learning Models

7. Training, Testing, and Validation

8. Thoughts

Search keywords: Neural computer Program induction Reinforcement learning Reinforcement learning Deepmind openai Vinyals pointer network

The project has the following steps:

1. Perform a literature review on the topic of marine surface vessel modeling and control, as well as deep learning techniques used in tandem with control problems.
2. Set up a simulation environment for the surface vessel, incorporating difference levels of disturbances.
3. Set up a baseline using the established control algorithms
4. Apply reinforcement learning to the system and figure out how to train the system effectively.
5. Analyse the performance of the machine learning controllers, compare the computation time (the expensiveness) between the two algorithms, and explore the stability consequences of using machine learning techniques in control.