Robotics and Autonomous Systems 84 (2016) 1–14

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Neural inverse reinforcement learning in autonomous navigation

Chen Xia *, Abdelkader El Kamel

Research Center CRIStAL, UMR CNRS 9189, École Centrale de Lille, 59651 Villeneuve d'Ascq, France

HIGHLIGHTS

- A novel model-free algorithm, Neural Inverse Reinforcement Learning, is introduced in performing autonomous navigation tasks.
- A nonlinear neural policy representation is proposed to establish the mapping between the state and action spaces.
- Computer-based expert demonstrations are supplemented to learning robots when human experts are not available in some extreme navigation tasks.
- A refinement operation based on maximum a posteriori is designed to pretreat the demonstrations from suboptimal experts.
- This method can easily deal with large state spaces and generalize unvisited states in demonstrations.

ARTICLE INFO

Article history:
Received 21 October 2015
Accepted 10 June 2016
Available online 23 June 2016

Keywords:
Inverse reinforcement learning
Learning from demonstration
Neural network
Autonomous navigation
Markov decision processes
Dynamic environments

ABSTRACT

Designing intelligent and robust autonomous navigation systems remains a great challenge in mobile robotics. Inverse reinforcement learning (IRL) offers an efficient learning technique from expert demonstrations to teach robots how to perform specific tasks without manually specifying the reward function. Most of existing IRL algorithms assume the expert policy to be optimal and deterministic, and are applied to experiments with relatively small-size state spaces. However, in autonomous navigation tasks, the state spaces are frequently large and demonstrations can hardly visit all the states. Meanwhile the expert policy may be non-optimal and stochastic. In this paper, we focus on IRL with large-scale and high-dimensional state spaces by introducing the neural network to generalize the expert's behaviors to unvisited regions of the state space and an explicit policy representation is easily expressed by neural network, even for the stochastic expert policy. An efficient and convenient algorithm, Neural Inverse Reinforcement Learning (NIRL), is proposed. Experimental results on simulated autonomous navigation tasks show that a mobile robot using our approach can successfully navigate to the target position without colliding with unpredicted obstacles, largely reduce the learning time, and has a good generalization performance on undemonstrated states. Hence prove the robot intelligence of autonomous navigation transplanted from limited demonstrations to completely unknown tasks.

1. Introduction

A rapid development of the robotic techniques has witnessed the advent of modern robots in large numbers. More and more robots are designed and produced to assist or replace our human beings to perform complicated control operations and planning tasks among plenty of domains. However, we are aware of the fact that designing reliable control systems for autonomous robot tasks is usually a complicated process, even for people specialized in programming robots. The number of uncertain situations which a robot may face and the wide spectrum of behaviors it may have to perform make the job of robot programming rather difficult. This sort of manually programming is generally an expensive as well as intense time-consuming process. Rather than pre-programming a robot for all the behaviors, it would be more useful if the robot could learn such tasks by themselves.

Recent researches has brought machine learning approaches into the community of robotics in order to improve the robot autonomous ability based on accumulated experiences. These artificial intelligent methods can be computationally less expensive than classical ones and can largely ease the burden on human designers.

In this paper, we are interested in designing intelligent mobile robots. The autonomous navigation ability is thus one of the fundamental skills, which requires a robot to traverse from some start location to a goal location in the meanwhile to avoid any potential collisions. Along the path, it must maintain knowledge

* Corresponding author.
E-mail addresses: chen.xia@centraliens-lille.org (C. Xia), abdelkader.elkamel@ec-lille.fr (A. El Kamel).

http://dx.doi.org/10.1016/j.robot.2016.06.003
0921-8890/

of its own position, observe the surrounding environment, plan actions to achieve its goal, and finally execute these actions. All decisions must be made in real time, and the robot may be faced with situations beyond the imagination of human designers. This makes it impractical to manually devise a policy for all the possible tasks that the robot may have to perform.

Machine learning techniques has been successfully applied to the state-of-the-art self-driving cars. Thrun et al. provided a comprehensive survey of Stanley, the winning robot of the 2005 DARPA Grand Challenge [1]. This robotic car was a milestone in the quest for self-driving cars. The pervasive use of machine learning techniques in outdoor obstacle detection and terrain mapping, both ahead and during the race, made Stanley robust and precise. However, since the race environment was static, Stanley was unable to navigate in traffic. Two years later, Junior, a robotic vehicle capable of navigating urban in-traffic environments autonomously was developed and won second place in the 2007 DARPA Urban Challenge [2]. The robot successfully demonstrated merging, intersection handling, parking lot navigation, lane changes and autonomous U-turns.

Reinforcement learning (RL) [3] is one of the machine learning methods that offers a powerful tool for constructing adaptive and intelligent systems. In the framework of RL, the learner is a decision-making agent that takes actions in an environment and receives an reinforcement signal for its actions in trying to accomplish a task. The signal, well known as reward (or penalty), evaluates an action's outcome, and the agent seeks to learn to select a sequence of actions, i.e. a policy, that maximize the total accumulated reward over time. Significant applications of reinforcement learning to enable the learning ability of autonomous systems can be found in [4–8]. Kober et al. also summarized the reinforcement learning techniques applied in robotics in [9].

The goal of a reinforcement learning agent is to collect as many rewards as possible, and an informative reward function becomes a fundamental assumption that a successful RL algorithm counts on. This type of the evaluation of robot behaviors always needs to be provided beforehand. However, in practice, defining the reward function can itself be a challenge because an informative reward function may be very hard to specify and exhaustive to tune for large and complex problems [10]. Inverse Reinforcement Learning (IRL) arose due to the curiosity of if a learning agent can discover the underlying rewards from a bunch of demonstrated examples of a desired behavior.

Rather than directly mimicking the expert with some supervised learning approach, IRL consists in learning a reward function under which the policy demonstrated by the expert is optimal. IRL is originally introduced in [11], where the authors addressed three learning problems: IRL in finite state spaces, IRL in infinite state spaces and IRL from sampled trajectories. In practice, it is easier to get samples from an expert. However, the authors also noted that the IRL problem is ill-posed. In fact, there exists a series of reward functions, including constant functions, that may lead to the same optimal policy. Abbeel and Ng then introduced a new indirect learning approach, named apprenticeship learning [12], where the learning is less concerned about the actual reward function, and the objective is to recover a policy that is close to the demonstrated behavior. It is assumed that the reward is a sum of weighted state features, and finds a reward function to match the demonstrator's feature expectations. This method may not explicitly recover the expert's reward function, but still output a policy that attains the performance close to that of the expert. This method is then implemented in helicopter control [13].

The maximum margin planning (MMP) algorithm [14] uses similar ideas, a linearized-features reward, where the learner attempts to find a policy that make the provided demonstrations look better than other policies by a margin, and minimizes a cost function between observed and predicted actions by a subgradient descent. Ratliff et al. extends the maximum margin planning and developed the LEARCH algorithm [15], and applied it to outdoor autonomous navigation. The idea of MMP also inspired the structured classification based inverse reinforcement learning (SCIRL) [16,17], where the authors use only sampled trajectories to reduce IRL to a structured classification problem and do not need to solve the direct RL process that many existing IRL algorithms require. Similarly, the dynamic policy programming was adopted in [18] in order to estimate the reward and the state value function without solving the MDP.

The Bayesian inverse reinforcement learning approaches [19,20] use probability distribution to tackle with the ill-posed problem. They assume that the demonstrator samples state–action sequences from a prior distribution over possible reward functions, and calculates a posterior on the reward function using Bayesian inference.

Similar to Bayesian IRL, the maximum entropy algorithm [21] use an MDP model for calculating a probability distribution on the state-action pairs. Maximum entropy IRL focuses on the distribution over trajectories rather than pure actions. Later on, based on the maximum entropy framework, the relative entropy inverse reinforcement learning algorithm using policy iteration is proposed in [22]. It indirectly employs knowledge of the environment and minimizes the relative entropy between the empirical distribution of the trajectories under a baseline policy and the distribution of the trajectories under a policy that matches the reward features of the demonstrations. A stochastic gradient descent is used to minimize the relative entropy.

Qiao and Beling proposed a Gaussian processes model and use preference graphs to represent observations of decision trajectories [23]. Levine et al. present a probabilistic algorithm for nonlinear inverse reinforcement learning and they use Gaussian process model to learn the reward as a nonlinear function [24].

Our research focuses on improving robot learning ability in autonomous navigation tasks via inverse reinforcement learning. An autonomous navigation task is a typical large-scale state-space problem. The demonstrators can only cover a small subset of the state spaces, and thus solving the generalization of state space is a key issue. Most inverse reinforcement learning algorithms use a linear feature-based state representations instead of directly using states in order to generalize on undemonstrated states and do not give an explicit policy representation in large-scale spaces.

In this paper, we present an efficient and convenient learning algorithm, neural inverse reinforcement learning (NIRL), and apply it to autonomous robot navigation tasks. We represent the states using a linear combination of state and action features and adopt an artificial neural network to generalize the expert's actions to unvisited regions of the state space. By this means, we propose an explicit nonlinear policy representation. The maximum margin method is applied to learn the reward function. The IRL algorithms in the literature generally assume that a model of the transition is known, which is unrealistic in an unpredictable navigation problem. Our method, on the contrary, is model-free. Experiments are conducted on simulated autonomous robot navigation, and the results show that a mobile robot using our approach can successfully accomplish an autonomous navigation task without colliding with unpredicted obstacles. NIRL largely reduces the learning time, and has a good generalization performance on undemonstrated states. Therefore, NIRL is proved to be a reliable and robust learning algorithm for endowing the mobile robots the autonomy and the intelligence.

The rest of the paper is organized as follows. The next section describes the fundamental background that we use in this paper. Section 3 describes the autonomous mobile robot model. The proposed nonlinear neural policy representation is given in