Available online at www.sciencedirect.com
ScienceDirect
Robotics and Autonomous Systems 54 (2006) 887–897
Robotics and Autonomous Systems
www.elsevier.com/locate/robot

Parametric POMDPs for planning in continuous state spaces

Alex Brooks*, Alexei Makarenko, Stefan Williams, Hugh Durrant-Whyte

ARC Centre of Excellence for Autonomous Systems, Australian Centre for Field Robotics, The University of Sydney, Sydney, NSW 2006, Australia

Received 15 October 2005; received in revised form 12 May 2006; accepted 24 May 2006
Available online 2 August 2006

Abstract

This work addresses the problem of decision-making under uncertainty for robot navigation. Since robot navigation is most naturally represented in a continuous domain, the problem is cast as a continuous-state POMDP. Probability distributions over state space, or beliefs, are represented in parametric form using low-dimensional vectors of sufficient statistics. The belief space, over which the value function must be estimated, has dimensionality equal to the number of sufficient statistics. Compared to methods based on discretising the state space, this work trades the loss of the belief space's convexity for a reduction in its dimensionality and an efficient closed-form solution for belief updates. Fitted value iteration is used to solve the POMDP. The approach is empirically compared to a discrete POMDP solution method on a simulated continuous navigation problem. We show that, for a suitable environment and parametric form, the proposed method is capable of scaling to large state-spaces.

Keywords: Partially observable Markov decision processes; Mobile robot navigation; Acting under uncertainty

1. Introduction

A Markov decision process (MDP) models the repeated interaction of an agent with a stochastic environment [1]. MDP-based approaches to planning are well-studied and effective in domains where perfect knowledge of the state of the world is available. Unfortunately they are less effective in problems where the state is uncertain, a condition which prevails in many real-world problems.

When the state is unknown but some (uncertain) information about the state is available through observations, the world can be described by a partially observable Markov decision process (POMDP) [2]. A POMDP model defines a probabilistic representation of an agent's world. Specifically, given an initial state and action, it defines probability distributions over possible resultant states and observations. Given a reward function, an agent's task is to select actions which maximise its expected sum of (possibly discounted) future rewards.

The POMDP task is challenging because the agent must consider both the history of all previous observations and actions, and the space of all possible future observations and actions. The task is simplified by the fact that, given knowledge of the POMDP model, the agent can maintain a probability distribution over states which summarises the entire history [3]. This distribution is usually referred to as the agent's belief. Maintaining a consistent belief allows the problem to be converted from a POMDP over partially observable states to an MDP over fully observable beliefs. Traditional MDP solution methods can then be applied to the resultant belief-state MDP [3,1].

A number of POMDP solution methods, including the one proposed in this paper, solve the resultant MDP using value iteration. Essentially, value iteration iteratively builds a value function which specifies the expected sum of discounted future rewards attainable from each belief-state. Given a value function, an agent can act by simply choosing the action which instantaneously maximises its value, which is equivalent to planning ahead.

The problem of robot navigation is often cast as a POMDP, on the grounds that localisation is inherently imperfect and MDP-based approaches do not account for this uncertainty. The POMDP solution explicitly models the robot's position uncertainty, making decisions based on the probabilistic distribution over pose space. This naturally imparts the useful

* Corresponding author.
E-mail addresses: a.brooks@cas.edu.au (A. Brooks), a.makarenko@cas.edu.au (A. Makarenko), s.williams@cas.edu.au (S. Williams), hugh@cas.edu.au (H. Durrant-Whyte).

0921-8890/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2006.05.007

property that the robot will trade off actions that move the robot towards its goal with actions that reduce the robot's uncertainty in a principled way.

The majority of value-based POMDP research for robot navigation has focussed on the discrete case, dividing configuration spaces into finite numbers of cells. Robot navigation, however, is a fundamentally continuous problem that is poorly represented in the discrete domain unless the discretisation is sufficiently fine. Discrete POMDP solution methods have problems with fine discretisations because the dimensionality of the belief space is equal to the number of states, and computational complexity increases rapidly with the dimensionality of the belief space.

This paper presents an approach to solving robot-navigation POMDP problems efficiently in continuous state spaces. We refer to this approach as a parametric POMDP solution method [4]. By constraining distributions over state space to a parametric family, points in the infinite-dimensional continuous belief space can be represented by finite vectors of sufficient statistics. Choosing a parametric family with a relatively small number of sufficient statistics results in a relatively low-dimensional belief space. For a given combination of MDP model and parametric form, it may be possible to find a (possibly approximate) belief transition function which preserves that parametric form. If so, belief updates can be performed efficiently, directly in the low-dimensional parameter-space. Since the value function is not likely to be piecewise-linear and convex (PWLC) in sufficient-statistic space, fitted value iteration [5] is used to solve the POMDP. For reasons described in Section 3, we focus on the use of Gaussian distributions as a parametric form.

The remainder of this paper is organised as follows. Section 2 discusses related approaches, and Section 3 formulates the dynamic programming equations on which the POMDP solution is founded and discusses the implications of a parametric representation. Section 4 describes a solution using this representation, Section 5 applies this solution to a robot navigation problem and Section 6 concludes and provides directions for future work.

2. Related work

This section provides a brief review of prior work on POMDP solution methods. For a more thorough review, readers are directed to [6] and the references therein.

2.1. Value-based solution methods

While this paper focusses on the continuous worlds found in the robotics domain, a wealth of discrete value-based solution methods can be applied by first discretising the world. Discrete value-based POMDP solution methods can be broadly categorised as either being gradient based or being based on fitted value iteration.

2.1.1. Gradient-based solution methods

Gradient-based solution methods exploit the fact that, for discrete state, observation and action spaces, the value function is piecewise-linear and convex (PWLC). It can therefore be represented by the supremum of a finite number of hyperplanes over the belief simplex, where each hyperplane is represented by an α-vector [7]. The major problem for gradient-based methods is that the order of the number of α-vectors required to represent the value function exactly is double-exponential in the planning horizon [6]. Thus, algorithms for gradient-based value iteration need to keep the number of α-vectors low while still maintaining a good approximation of the true value function.

It is well-known that not all α-vectors contribute to the supremum, and therefore do not affect the value function. A number of algorithms perform exact value iteration by finding the minimal set of α-vectors required at each step, either by enumerating a superset of the required vectors and pruning the useless ones [8–10] or by iteratively expanding a subset until all vectors have been found [7,11,2]. For a more thorough review of exact algorithms, the reader is directed to [12].

Unfortunately, strategies for finding the minimal set of vectors to represent the value function exactly are usually computationally expensive and seem to make a difference only in the constant factors rather than the order of the growth [13]. As a result, exact algorithms are generally considered to be intractable for all but trivial problems.

Rather than generating all α-vectors required to represent the entire value function exactly, point-based algorithms perform approximate value iteration by generating only those vectors which maximise the value at a discrete set of belief points B [14,13,15–17,6]. It is hoped that the gradient information provided by the α-vectors will generalise well to other beliefs. The Perseus algorithm uses this gradient information to further reduce the required number of α-vectors [14]. At each iteration it finds a subset of belief points in B which, when updated, will improve the value at all B. As a recent discrete POMDP algorithm with available code, Perseus is compared to the algorithm proposed in this paper in Section 5.

Hoey et al. extend point-based value iteration to continuous observation spaces, using the fact that observations are useful only to the extent that they lead to different courses of action [18]. The observation space can therefore be partitioned by calculating the thresholds at which different observations require different actions. It is unclear how appropriate this is for robot navigation problems in which the action space is fundamentally continuous, and ideally every observation should lead to a different action.

2.1.2. Methods based on fitted value iteration

An alternative to gradient-based methods is to represent the value explicitly only at a discrete set of belief points B, and use a function approximator to represent the value between points. These approaches are often known as *grid-based* methods, however we find the term somewhat misleading because the beliefs in B need not exhibit any regular structure. Since the method proposed in this paper is based on fitted value iteration, the mechanics are described in more detail in Section 4.

A challenge for fitted value iteration techniques is the design of an efficient function approximator, since a discrete state space gives rise to a high dimensional belief space in which