ARTICLE IN PRESS

Engineering Applications of Artificial Intelligence ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Optimistic planning with an adaptive number of action switches for near-optimal nonlinear control

Koppány Máthé [a], Lucian Buşoniu [a,*], Rémi Munos [b], Bart De Schutter [c]

[a] Department of Automation, Technical University of Cluj-Napoca, Romania
[b] Google DeepMind, United Kingdom
[c] Delft Center for Systems and Control, Delft University of Technology, The Netherlands

ARTICLE INFO

Keywords:
Optimal control
Planning
Nonlinear predictive control
Near-optimality analysis

ABSTRACT

We consider infinite-horizon optimal control of nonlinear systems where the control actions are discrete, and focus on optimistic planning algorithms from artificial intelligence, which can handle general nonlinear systems with nonquadratic costs. With the main goal of reducing computations, we introduce two such algorithms that only search for constrained action sequences. The constraint prevents the sequences from switching between different actions more than a limited number of times. We call the first method optimistic switch-limited planning (OSP), and develop analysis showing that its fixed number of switches S leads to polynomial complexity in the search horizon, in contrast to the exponential complexity of the existing OP algorithm for deterministic systems; and to a correspondingly faster convergence towards optimality. Since tuning S is difficult, we introduce an adaptive variant called OASP that automatically adjusts S so as to limit computations while ensuring that near-optimal solutions keep being explored. OSP and OASP are analytically evaluated in representative special cases, and numerically illustrated in simulations of a rotational pendulum. To show that the algorithms also work in challenging applications, OSP is used to control the pendulum in real time, while OASP is applied for trajectory control of a simulated quadrotor.

1. Introduction

Optimal control problems arise in numerous areas of technology. Our focus here is on optimal control in discrete time, so as to maximize a discounted sum of rewards (negative costs). *Optimistic planning* (OP) techniques (Munos, 2014) solve this problem locally for any given state, by exploring tree representations of possible sequences of actions (control inputs) from that state, where the tree depth of each sequence is equal to its length. Given a computational budget of tree node expansions, performance grows with the resulting depth of the tree, which can be seen as an adaptive horizon. OP works for general dynamics and rewards, and provides a tight characterization of the relation between the computational budget and near-optimality. Motivated by these features, a number of OP algorithms have been introduced, e.g. by Kocsis and Szepesvári (2006), Bubeck and Munos (2010), Buşoniu and Munos (2012) and Mansley et al. (2011), which have proven useful in practical problems (Mansley et al., 2011; Gelly et al., 2006). OP is usually applied online in receding horizon, as a type of model-predictive control (MPC).

In this paper, we consider deterministic systems with discrete (or discretized) actions, and introduce two new OP techniques tailored for sequences that are constrained to switch only a limited number of times between different discrete actions. Inheriting the generality of OP, these techniques are able to deal with nonlinear dynamics and nonquadratic reward functions. The switch constraint is motivated by two classes of problems. In the first class (i), the loss of performance induced by the constraint is negligible—such as in time-optimal control, where solutions are of the bang–bang type. In the second class (ii), the switch constraint must be imposed due to the problem's nature, accepting the resulting performance degradation—for example, to decrease computation time or because setting the actuator to a new discrete level is costly. Examples of the latter type include traffic signal control (De Schutter and De Moor, 1998), water level control by barriers and sluices (van Ekeren et al., 2013), networked control systems (Tabuada, 2007), etc.

First, we propose *optimistic switch-limited planning* (OSP): an algorithm that only explores sequences with at most S switches, with S fixed. This allows a significant reduction in computational complexity

* Corresponding author.
E-mail addresses: koppany.mathe@aut.utcluj.ro (K. Máthé), lucian.busoniu@aut.utcluj.ro (L. Buşoniu), munos@google.com (R. Munos), b.deschutter@tudelft.nl (B. De Schutter).

https://doi.org/10.1016/j.engappai.2017.08.020
Received 24 November 2016; Received in revised form 7 June 2017; Accepted 29 August 2017
Available online xxxx
0952-1976/

with respect to the state-of-the-art OP algorithm in the discrete-action, deterministic case: OP for deterministic systems (OPD) (Hren and Munos, 2008). Indeed, we show that the computational effort needed by OSP to reach a given depth in the tree is polynomial in this depth, rather than exponential as in OPD. Therefore, given a computational budget n, the tree depth grows quickly and OSP converges faster to the switch-constrained optimal solution than OPD would converge to the unconstrained one. The convergence rate is dictated by the degree of the polynomial, a complexity measure for the optimal control problem.

A limitation of OSP is the need to manually tune the number of switches S. A too small value can lead to suboptimal solutions, while allowing too many switches may lead to unneeded computation. We therefore develop optimistic *adaptive* switch-limited planning (OASP), which automatically finds a good S. The value of S is increased adaptively, exploring sequences with more action switches when indicated by an increment rule. We illustrate two such rules, and analyze both variants in the same special cases as OSP was analyzed.

OSP is applied in receding horizon simulations to the problem of swinging up a rotational pendulum. Note that this problem is in class (i) where near-optimal sequences switch rarely. To illustrate class (ii), in particular systems where switches are costly, we show how OSP can take into account bandwidth limitations in networked control systems. Here, the constraint is enforced in closed loop, so that along any range of N consecutive steps there are at most S switches, where N is a parameter. Furthermore, OSP is applied to control the physical pendulum in real time. To evaluate the second algorithm, OASP, it is compared to OPD and OSP in simulations of the rotational pendulum, showing that in certain cases OASP performs better than the other methods, while remaining competitive in other cases. Finally, OASP is applied to the more complex control problem of trajectory control for quadrotors, showing the benefits of the novel algorithm over OPD and over a classical linear–quadratic regulator.

Like the entire class of OP algorithms, OSP and OASP are related to Monte Carlo tree search (Browne et al., 2012), heuristic search (Edelkamp and Schrödl, 2012), and planning for robotics (La Valle, 2006). The complexity measure of OSP (polynomial degree) is related to similar measures in other optimistic algorithms, e.g. the branching factor of near-optimal sequences in OPD (Hren and Munos, 2008), the near-optimality exponent in the stochastic case (Buşoniu and Munos, 2012), or the near-optimality dimension in optimization (Munos, 2014); due to the different structure of the explored tree, these measures do not work in the switch-constrained problem of OSP, and the new polynomial degree is needed.

In the MPC field, similar constraints on the number of action changes have been exploited to decrease computation, e.g. in the linear case by De Schutter and De Moor (1998), De Schutter (2000) and Alende et al. (2009), later extended to the nonlinear case as time-instant optimization MPC (van Ekeren et al., 2013). Applications include hybrid control (De Schutter and De Moor, 1998; Martinez et al., 2007; Alende et al., 2009) and hierarchical control (Sadowska et al., 2013). Liu et al. (2011) constrain the solutions to hold the command constant for a preset number of steps. In these works, an off-the-shelf optimizer (e.g. of the mixed-integer linear programming type, see Alves and Clímaco, 2007) is usually applied, and the computational effort is investigated empirically. Compared to this, the main advantage of our approach is an analytical characterization of the relationship between the computational effort and near-optimality, for the complete algorithm down to the implementation of the optimizer. A second axis of related work in MPC concerns complexity analysis, typically for linear–quadratic problems, see e.g. Li and Marlin (2011). A particularly strong work thread is in explicit MPC (Bemporad et al., 2002), where the optimal state feedback law is piecewise affine and the complexity of the online search for the current affine region is characterized, see e.g. Tøndel et al. (2003), Wen et al. (2009) and Bayat et al. (2011). Overall, MPC typically uses a fixed, finite horizon, and its main strengths include stability guarantees, mechanisms to handle constraints, and output feedback techniques. In contrast, OSP and OASP focus on the generality of the nonlinear dynamics they can address, while providing near-optimality and convergence rate guarantees with respect to the infinite-horizon optimum.

This paper is a revised and extended version of our conference article (Mathe et al., 2014), where OSP was introduced. The present paper provides more details and insight into the analysis of OSP, while its empirical evaluation is done using a different control problem, with entirely new real-time results. The main novelty compared to Mathe et al. (2014) is however the adaptive algorithm OASP, with its analysis, numerical evaluation, and application to simulated quadrotor trajectory control.

Next, Section 2 gives the necessary background, Section 3 introduces and analyzes OSP, and Section 4 similarly presents and studies OASP. Experimental results for the two methods are provided in Sections 5 and 6, respectively. Finally, Section 7 concludes the paper.

2. Background: Markov decision processes and optimistic planning for deterministic systems

Consider a Markov decision process (MDP) describing an optimal control problem with state $x \in X$, action $u \in U$, transition function $f : X \times U \to X, f(x,u) = x'$ and an associated reward function $\rho : X \times U \to \mathbb{R}$. The function $f(x,u)$ describes the transition from state x to x' when applying action u, i.e. the system dynamics. Each transition is rewarded by $\rho(x,u)$.

We assume that the action space U is finite and discrete, $U = \{u^1, \ldots, u^M\}$, and the system dynamics $f(x,u)$ and the reward function $\rho(x,u)$ are known. Additionally, to facilitate the analysis, the reward function is assumed to be bounded to the unit interval, $\rho(x,u) \in [0,1], \forall x,u$. The only restrictive part here is the boundedness of the reward, which is often assumed in AI approaches to solving MDPs; then, the rewards can be scaled and translated to the unit interval without affecting the optimal solution.

The objective is to find for any given state x_0 an infinite action sequence $h_\infty = (u_0, u_1, \ldots)$ that maximizes the value function (discounted sum of rewards):

$$v(h_\infty) = \sum_{k=0}^{\infty} \gamma^k \rho(x_k, u_k) \tag{1}$$

where $k \geq 0$ is the discrete time step, $x_{k+1} = f(x_k, u_k)$, and $\gamma \in (0,1)$ is the discount factor. The optimal value is denoted by $v^* = \sup_{h_\infty} v(h_\infty)$.

Optimistic Planning for Deterministic Systems (OPD) (Hren and Munos, 2008; Munos, 2014) is an extension of the classical A^* tree search to infinite-horizon problems. OPD looks for v^* by creating a search tree starting from x_0 that explores the space of action sequences by simulating their effects, until a given computational budget is exhausted. This budget is denoted by n and measures the number of nodes the algorithm is allowed to expand in the search tree, where expanding a node means adding M child nodes to it, one corresponding to each action from U. Fig. 1 shows an example of a tree after $n = 3$ expansions have been performed.

A node at depth d is equivalent to the action sequence $h_d = (u_0, u_1, \ldots, u_{d-1})$ leading to it: e.g. in Fig. 1, for the bold node at depth $d = 3$ one has $h_3 = (u^1, u^2, u^2)$. Consider any infinitely long action sequence h_∞ that starts with h_d. Now, define the following lower bound on $v(h_\infty)$:

$$\nu(h_d) = \sum_{k=0}^{d-1} \gamma^k \rho(x_k, u_k) \leqslant v(h_\infty) \tag{2}$$

and the following upper bound on $v(h_\infty)$:

$$b(h_d) = \nu(h_d) + 1 \cdot \gamma^d + 1 \cdot \gamma^{d+1} + \cdots = \nu(h_d) + \frac{\gamma^d}{1-\gamma} \geqslant v(h_\infty). \tag{3}$$

Note that these bounds are valid because $\gamma < 1$ and the rewards take values between 0 and 1.

Please cite this article in press as: Máthé K., et al., Optimistic planning with an adaptive number of action switches for near-optimal nonlinear control. Engineering Applications of Artificial Intelligence (2017), https://doi.org/10.1016/j.engappai.2017.08.020.