European Journal of Operational Research 211 (2011) 343–351

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Stochastics and Statistics

Approximate dynamic programming via direct search in the space of value function approximations

E.F. Arruda [a,*], M.D. Fragoso [b], J.B.R. do Val [c]

[a] FENG/PUCRS. Av. Ipiranga, 6681. Porto Alegre, RS 90619-900, Brazil
[b] CSC/LNCC. Av. Getúlio Vargas, 333. Petrópolis, RJ 25651-075, Brazil
[c] DT/FEEC/UNICAMP. Cidade Universitária Zeferino Vaz. Av. Albert Einstein, 400. CP 6101. Campinas, SP 13083-852, Brazil

ARTICLE INFO

Article history:
Received 4 January 2010
Accepted 17 November 2010
Available online 21 November 2010

Keywords:
Dynamic programming
Markov decision processes
Convex optimization
Direct search methods

ABSTRACT

This paper deals with approximate value iteration (AVI) algorithms applied to discounted dynamic programming (DP) problems. For a fixed control policy, the span semi-norm of the so-called Bellman residual is shown to be convex in the Banach space of candidate solutions to the DP problem. This fact motivates the introduction of an AVI algorithm with local search that seeks to minimize the span semi-norm of the Bellman residual in a convex value function approximation space. The novelty here is that the optimality of a point in the approximation architecture is characterized by means of convex optimization concepts and necessary and sufficient conditions to local optimality are derived. The procedure employs the classical AVI algorithm direction (Bellman residual) combined with a set of independent search directions, to improve the convergence rate. It has guaranteed convergence and satisfies, at least, the necessary optimality conditions over a prescribed set of directions. To illustrate the method, examples are presented that deal with a class of problems from the literature and a large state space queueing problem setting.

1. Introduction

The rise of dynamic programming (DP) (Bellman, 1957) was a major breakthrough in the treatment and solution of deterministic and stochastic sequential decision problems. There is nowadays a wide variety of applications for this framework (Puterman, 1994), ranging from scheduling problems, e.g. (Sox et al., 1999) to complex and network models as presented by Swarts and Ferreira (1993), Meyn (2008). The elegant DP recursion is efficient because it allows an implicit comparison between a combinatorial number of scenarios by enumerating the *states*, i.e. possible configurations, of the system. However, the number of states in a system increases exponentially with the system dimension, thus making standard DP algorithms prohibitively demanding for problems with a moderately large number of dimensions. Detailed treatments of DP techniques can be found in the classical works by Puterman (1994) and Bertsekas (1995). For an interesting study on the convergence properties of standard dynamic programming algorithms we refer to Zobel and Scherer (2005). A related study on the effectiveness of action elimination in value iteration algorithms was conducted in Jaber (2008).

The approximate dynamic programming (ADP) framework comprises a body of theory and computational tools developed to tackle situations where standard DP algorithms become computationally too demanding, see for example (Bertsekas and Tsitsiklis, 1996; Si et al., 2004; Sutton and Barto, 1998, or Powell, 2007). For further results and applications of ADP methods, we refer to Bertsekas and Yu (2009), Yu and Bertsekas (2009), Borkar et al. (2009), Choi and Van Roy (2006), Menache et al. (2005). A popular ADP approach involves incorporating an arbitrary parametric approximation scheme into the original DP problem and consists in seeking sub-optimal solutions in a lower dimensional subset of the standard value iteration algorithm search space. Although this approach has proven successful in real-world applications (e.g. Tesauro, 1992), some measure of refinement is needed, for it may lead to unstable and possibly divergent algorithms (Boyan and Moore, 1995). It was the possibility of erratic behavior that led to the development of convergent ADP algorithms specifically tailored for certain types of approximation schemes (architectures), e.g. (Baird, 1995; Gordon, 1995).

Convergent ADP algorithms often rely on specific properties of the approximation architecture and/or the *projection/fitting operator*, i.e. the operator that converts elements in the Banach space of value function candidates into elements (*approximate solutions*) in the approximation space. The non-expansion based algorithm introduced in Gordon (1995) applies to non-expansive projection

* Corresponding author. Tel.: +55 51 3353 4403; fax +55 51 3320 3625.
E-mail addresses: efarruda@ieee.org (E.F. Arruda), frag@lncc.br (M.D. Fragoso), jbosco@dt.fee.unicamp.br (J.B.R. do Val).

0377-2217/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2010.11.019

mappings. Residual and gradient descent algorithms, e.g. (Baird, 1995; Baird and Moore, 1999) perform gradient descent search with respect to the mean squared Bellman residual, thereby requiring that each value function in the approximation space be differentiable with respect to the parameters, i.e. requiring the existence of a differential mapping from approximate value functions to parameters. Although convergence is guaranteed, not much can be inferred about the nature of the accumulation point. A convergent algorithm for fairly general approximation architectures under a class of expansive projection mappings was introduced in Arruda and do Val (2006). It was shown to converge to the projection of a local solution to the DP problem in the approximation space under suitable conditions.

A peculiar feature of the present paper is that it introduces a class of ADP algorithms with value function approximation that seeks to minimize the span semi-norm of the Bellman residual. The rationale behind the proposed approach is similar to that behind the residual algorithms of Baird (1995) and Baird and Moore (1999), namely, to find a low residual approximate solution while relying on the fact that the Bellman residual is an estimate of the distance between any value function candidate and the *true* value function. Nevertheless, whereas the main concern of residual algorithms is to ensure convergence by applying a gradient descent procedure, the approach proposed here, while also ensuring convergence, is additionally concerned with unveiling the nature of the accumulation point. The span semi-norm is chosen, instead of the generally applied supremum norm, to account for the fact that adding a constant to an approximate value function does not alter the corresponding *greedy control policy* and, consequently, the quality of the approximation (Bertsekas, 1995). Accordingly, the span semi-norm of both an approximate value function and its Bellman residual remain unchanged with the addition of a constant scalar (Bertsekas, 1995; Puterman, 1994).

A novel contribution of this work is to prove that, for a fixed control policy, an appropriate function of the Bellman residual, namely the span semi-norm, is convex in the Banach space of real-valued functions. Such a conclusion implies the existence of a single global optimizer for that function under a convex approximation architecture. In addition, convex programming theory, e.g. (Bazaraa et al., 1993), is applied to derive necessary and sufficient local optimality conditions, enabling the user to unveil the nature of the accumulation point. This motivates the introduction of an ADP algorithm with local search and guaranteed convergence. The algorithm seeks the minimum Bellman residual in the approximation space and converges, at a minimum, to an accumulation point that satisfies necessary optimality conditions in a set of prescribed directions within the approximation space.

Another interesting and distinguishing feature of the proposed ADP algorithm is that it is inspired by direct search (Lewis et al., 2000) and derivative free unidimensional search (Bazaraa et al., 1993) optimization procedures and does not make use of gradient information. Therefore, in contrast to residual gradient algorithms, the proposed algorithm does not require the existence of a differential mapping from approximate value functions to parameters. As a result, the proposed approach can be applied to fairly general convex approximation architectures and can be viewed as a generalization of the residual gradient approach. Preliminary results of the present study were presented in Arruda et al. (2008).

This paper is organized as follows. Section 2 gives a general description of exact and approximate discounted dynamic programming problems. Section 3 presents the proposed formulation of the ADP problem. Section 4 investigates convexity properties of the proposed objective function and characterizes local optimality. A pair of ADP algorithms that incorporates concepts of convex optimization and direct search methods is presented in Section 5. Numerical experiments are presented in Sections 6 and 7 concludes the paper.

2. Preliminaries

Consider a discrete dynamic programming problem P whose controlled dynamics are described by a Markov chain X_k, $k \geqslant 0$. Let S denote the state space of the problem and $U(x)$, $x \in S$ denote the set of available control actions at state x. At any period k, with $X_k = x \in S$, a control action $u \in U(x)$ is taken, an instantaneous non-negative cost $c(x,u)$ is incurred and the system moves to some state $y \in S$ with probability p^u_{xy}. Note that this formulation encompasses the class of deterministic problems, for which we have $p^u_{xy} = 1$ for some $y \in S$ and nil otherwise.

A stationary deterministic control policy $\pi : S \to U$ is a mapping that prescribes a single control action $u = \pi(x)$ to be taken each time the system visits state x. Let Π be the class of feasible stationary policies and $\alpha \in (0,1)$ be a discount factor. Associated to any policy $\pi \in \Pi$ is a discounted long term cost

$$V^\pi(x) = E_\pi\left[\sum_{k=0}^{\infty} \alpha^k c(X_k, \pi(X_k))\right], \quad X_0 = x, \tag{1}$$

where E_π denotes the conditional expectation given that policy π is applied. The objective of solving problem P is to find a stationary policy π^* such that

$$V^*(x) := V^{\pi^*}(x) \leqslant V^\pi(x), \quad \forall \pi \in \Pi \text{ and } x \in S.$$

Such a policy exists and is unique whenever $U(x)$ is finite for each $x \in S$ and S is countable (Puterman, 1994, Theorem 6.2.10). For more general conditions for the existence of an optimal stationary policy, we refer to Harrison (1972) or Bertsekas and Shereve (2007).

Let $\mathbb{V}$ be the space of non-negative real valued functions $V : S \to \mathbb{R}$ and define mappings $T^u : \mathbb{V} \to \mathbb{V}$ and $T : \mathbb{V} \to \mathbb{V}$, respectively, such that for each $x \in S$ and $u \in U(x)$,

$$T^u V(x) := c(x,u) + \alpha E_u[V(X_1)|X_0 = x] = c(x,u) + \alpha \sum_{y \in S} p^u_{xy} V(y), \tag{2}$$

$$TV(x) = \min_{u \in U(x)} T^u V(x). \tag{3}$$

Standard DP theory states that T is a contraction mapping with respect to the supremum norm, denoted in this paper as $\|\cdot\|_\infty$, and its unique fixed point (V^*) coincides with the solution to the problem P. Moreover, V^* can be computed recursively by the value iteration (VI) algorithm

$$\begin{aligned} &V_0 \in \mathbb{V}, \\ &V_{k+1} = TV_k. \end{aligned} \tag{4}$$

Definition 1. Let A be a subset of $\mathbb{V}$. For any function $f : A \to \mathbb{R}$, we say that $\delta := V_2 - V_1$, $V_1, V_2 \in A$, is a descent direction if $f(V_2) \leqslant f(V_1)$.

The recursion in (4) can be deemed as an unconstrained subgradient algorithm that takes at any iteration k a *descent direction* $d_k = TV_k - V_k$ with respect to the Bellman residual. Both the convergence and uniqueness of the fixed point of the recursion in (4) follow from the contraction property, that reads

$$\|TV - TV'\|_\infty \leqslant \alpha \|V - V'\|_\infty, \quad \forall V, V' \in \mathbb{V}. \tag{5}$$

When the state space is prohibitively large, an alternative is to substitute the recursion in (4) for an approximate value iteration (AVI) algorithm that seeks an approximate solution in a parametric *approximation space* $A \subset \mathbb{V}$. The approximate algorithm applies mapping T to a subset of S and *projects* the samples thus obtained