14 NEURO-DYNAMIC PROGRAMMING: OVERVIEW AND RECENT TRENDS

Benjamin Van Roy

Abstract: Neuro-dynamic programming is comprised of algorithms for solving large-scale stochastic control problems. Many ideas underlying these algorithms originated in the field of artificial intelligence and were motivated to some extent by descriptive models of animal behavior. This chapter provides an overview of the history and state-of-the-art in neuro-dynamic programming, as well as a review of recent results involving two classes of algorithms that have been the subject of much recent research activity: temporal-difference learning and actor-critic methods.

14.1 INTRODUCTION

In the study of decision-making, there is a dividing line between those who seek an understanding of how decisions *are made* and those who analyze how decisions *ought to be made* in the light of clear objectives. Among the former group are psychologists and economists who examine participants of physical systems in their full complexity. This often entails the consideration of both "rational" and "irrational" behavior. The latter group—those concerned with *rational decision-making*—includes engineers and management scientists who focus on the strategic behavior of sophisticated agents with definite purposes. The intent is to devise strategies that optimize certain criteria and/or meet specific demands. The problems here are well-defined and the goal is to find a "correct" way to make decisions, if one exists.

The self-contained character of rational decision problems has provided a ground for the development of much mathematical theory. Results of this work—as exemplified by previous chapters of this volume—provide an understanding of various possible models of dynamics, uncertainties, and objectives, as well as characterizations of optimal decision strategies in these settings. In cases where optimal strategies do exist, the theory is complemented by computational methods that deliver them.

In contrast to rational decision-making, there is no clear-cut mathematical theory about decisions made by participants of natural systems. Scientists are forced to propose speculative theories, and to refine their ideas through experimentation. In this context, one approach has involved the hypothesis that behavior is in some sense rational. Ideas from the study of rational decision-making are then used to characterize such behavior. In financial economics, this avenue has lead to utility and equilibrium theory. To this day, models arising from this school of economic thought—though far from perfect—are employed as mainstream interpretations of the dynamics of capital markets. The study of animal behavior presents another interesting case. Here, evolutionary theory and its popular precept—"survival of the fittest"—support the possibility that behavior to some extent concurs with that of a rational agent.

There is also room for reciprocal contributions from the study of natural systems to the science of rational decision-making. The need arises primarily due to the computational complexity of decision problems and the lack of systematic approaches for dealing with it. For example, practical problems addressed by the theory of dynamic programming can rarely be solved using dynamic programming algorithms because the computational time required for the generation of optimal strategies typically grows exponentially in the number of variables involved—a phenomenon known as the *curse of dimensionality.* This deficiency calls for an understanding of suboptimal decision-making in the presence of computational constraints. Unfortunately, no satisfactory theory has been developed to this end.

It is interesting to note that similar computational complexities arise in attempts to automate decision tasks that are naturally performed by humans or animals. The fact that biological mechanisms facilitate the efficient synthesis of adequate strategies motivates the possibility that understanding such mechanisms can inspire new and computationally feasible methodologies for strategic decision-making.

Over the past two decades, algorithms of *reinforcement learning*—originally conceived as descriptive models for phenomena observed in animal behavior–have grown out of the field of artificial intelligence and been applied to solving complex sequential decision problems. The success of reinforcement learning algorithms in solving large-scale problems has generated excitement and intrigue among operations researchers and control theorists, and much subsequent research has been devoted to understanding such methods and their potential. Developments have focused on a normative view, and to acknowledge the relative disconnect from descriptive models of animal behavior, some operations researchers and control theorists have come to refer to this area of research as *neuro-dynamic programming*, instead of *reinforcement learning.*

In this chapter, we provide a sample of recent developments and open issues at the frontier of research in neuro-dynamic programming. Our two points of focus are temporal-difference learning and actor-critic methods—two algorithmic ideas that have found greatest use in applications of neuro-dynamic programming and for which there has been significant theoretical progress in recent years. We begin, though, with three sections providing some background and perspective on the methodology and problems that may address.

14.2 STOCHASTIC CONTROL

As a problem formulation, let us consider a discrete-time dynamic system that, at each time t, takes on a state x_t and evolves according to

$$x_{t+1} = f(x_t, a_t, w_t),$$

where w_t is a disturbance and a_t is a control decision. Though more general (infinite/continuous) state spaces can be treated, to keep the exposition simple, we restrict attention to finite state, disturbance, and control spaces, denoted by $\mathbb{X}$, $\mathbb{W}$, and $\mathbb{A}$, respectively. Each disturbance $w_t \in \mathbb{W}$ is independently sampled from some fixed distribution.

A function $r : \mathbb{X} \times \mathbb{A} \mapsto \mathbb{R}$ associates a reward $r(x_t, a_t)$ with a decision a_t made at state x_t. A *stationary policy* is a mapping $\phi : \mathbb{X} \mapsto \mathbb{A}$ that generates state-contingent decisions. For each stationary policy ϕ, we define a value function $v(\cdot, \phi) : \mathbb{X} \mapsto \mathbb{R}$ by

$$v(x, \phi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t r(x_t, \phi(x_t)) \Big| x_0 = x\right],$$

where $\beta \in [0, 1)$ is a discount factor and the state sequence is generated according to $x_0 = x$ and $x_{t+1} = f(x_t, \phi(x_t), w_t)$. Each $v(x, \phi)$ can be interpreted as an assessment of long term rewards given that we start in state x and control the system using a stationary policy ϕ. The optimal value function V is defined by

$$V(x) = \max_{\phi} v(x, \phi).$$

A standard result in dynamic programming states that any stationary policy ϕ^* given by

$$\phi^*(x) = \operatorname*{argmax}_{a \in \mathbb{A}} \mathop{\mathbb{E}}_{w} \Big[r(x, a) + \beta V(f(x, a, w)) \Big],$$

where $\mathop{\mathbb{E}}_{w}[\cdot]$ denotes expectation with respect to the distribution of disturbances, is optimal in the sense that

$$V(x) = v(x, \phi^*),$$

for every state x (see, e.g. [8]).

For illustrative purposes, let us provide one example of a stochastic control problem.

Example 14.1 *The video arcade game of Tetris can be viewed as an instance of stochastic control (we assume that the reader is familiar with this popular game). In particular, we can view the state x_t as an encoding of the current "wall of bricks" and the shape of the current "falling piece." The decision a_t*