Journal of Machine Learning Research 9 (2008) 1679-1709 Submitted 6/07; Revised 12/07; Published 8/08

Value Function Based Reinforcement Learning in Changing Markovian Environments

Balázs Csanád Csáji BALAZS.CSAJI@SZTAKI.HU
László Monostori* LASZLO.MONOSTORI@SZTAKI.HU
Computer and Automation Research Institute
Hungarian Academy of Sciences
Kende utca 13–17, Budapest, H–1111, Hungary

Editor: Sridhar Mahadevan

Abstract

The paper investigates the possibility of applying value function based reinforcement learning (RL) methods in cases when the environment may change over time. First, theorems are presented which show that the optimal value function of a discounted Markov decision process (MDP) Lipschitz continuously depends on the immediate-cost function and the transition-probability function. Dependence on the discount factor is also analyzed and shown to be non-Lipschitz. Afterwards, the concept of (ε, δ)-MDPs is introduced, which is a generalization of MDPs and ε-MDPs. In this model the environment may change over time, more precisely, the transition function and the cost function may vary from time to time, but the changes must be bounded in the limit. Then, learning algorithms in changing environments are analyzed. A general relaxed convergence theorem for stochastic iterative algorithms is presented. We also demonstrate the results through three classical RL methods: asynchronous value iteration, Q-learning and temporal difference learning. Finally, some numerical experiments concerning changing environments are presented.

Keywords: Markov decision processes, reinforcement learning, changing environments, (ε, δ)-MDPs, value function bounds, stochastic iterative algorithms

1. Introduction

Stochastic control problems are often modeled by Markov decision processes (MDPs) that constitute a fundamental tool for computational learning theory. The theory of MDPs has grown extensively since Bellman introduced the discrete stochastic variant of the optimal control problem in 1957. These kinds of stochastic optimization problems have great importance in diverse fields, such as engineering, manufacturing, medicine, finance or social sciences. Several solution methods are known, for example, from the field of [neuro-]dynamic programming (NDP) or reinforcement learning (RL), which compute or approximate the optimal control policy of an MDP. These methods succeeded in solving many different problems, such as transportation and inventory control (Van Roy et al., 1996), channel allocation (Singh and Bertsekas, 1997), robotic control (Kalmár et al., 1998), production scheduling (Csáji and Monostori, 2006), logical games and problems from financial mathematics. Many applications of RL and NDP methods are also considered by the textbooks of Bertsekas and Tsitsiklis (1996), Sutton and Barto (1998) as well as Feinberg and Shwartz (2002).

∗. Also faculty in Mechanical Engineering at the Budapest University of Technology and Economics.

The dynamics of (Markovian) control problems can often be formulated as follows:

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

where x_t is the state of the system at time $t \in \mathbb{N}$, a_t is a control action and w_t is some disturbance. There is also a cost function $g(x_t, a_t)$ and the aim is to find an optimal control policy that minimizes the [discounted] costs over time (the next section will contain the basic definitions). In many applications the calculation of a control policy should be fast and, additionally, environmental changes should also be taken into account. These two criteria are against each other. In most control applications during the computation of a control policy the system uses a *model* of the environment. The dynamics of (1) can be modeled with an MDP, but what happens when the model is wrong (e.g., if the transition function is incorrect) or the dynamics have changed? The changing of the dynamics can also be modeled as an MDP, however, including environmental changes as a higher level MDP very likely leads to problems which do not have any practically efficient solution methods.

The paper argues that if the model was "close" to the environment, then a "good" policy based on the model cannot be arbitrarily "wrong" from the viewpoint of the environment and, moreover, "slight" changes in the environment result only in "slight" changes in the optimal cost-to-go function. More precisely, the optimal value function of an MDP depends Lipschitz continuously on the cost function and the transition probabilities. Applying this result, the concept of (ε, δ)-MDPs is introduced, in which these functions are allowed to vary over time, as long as the cumulative changes remain bounded in the limit.

Afterwards, a general framework for analyzing stochastic iterative algorithms is presented. A novelty of our approach is that we allow the value function update operator to be time-dependent. Then, we apply that framework to deduce an approximate convergence theorem for time-dependent stochastic iterative algorithms. Later, with the help of this general theorem, we show relaxed convergence properties (more precisely, κ-approximation) for value function based reinforcement learning methods working in (ε, δ)-MDPs.

The main contributions of the paper can be summarized as follows:

1. We show that the optimal value function of a discounted MDP Lipschitz continuously depends on the immediate-cost function (Theorem 12). This result was already known for the case of transition-probability functions (Müller, 1996; Kalmár et al., 1998), however, we present an improved bound for this case, as well (Theorem 11). We also present value function bounds (Theorem 13) for the case of changes in the discount factor and demonstrate that this dependence is not Lipschitz continuous.

2. In order to study changing environments, we introduce (ε, δ)-MDPs (Definition 17) that are generalizations of MDPs and ε-MDPs (Kalmár et al., 1998; Szita et al., 2002). In this model the transition function and the cost function may change over time, provided that the accumulated changes remain bounded in the limit. We show (Lemma 18) that potential changes in the discount factor can be incorporated into the immediate-cost function, thus, we do not have to consider discount factor changes.

3. We investigate stochastic iterative algorithms where the value function operator may change over time. A relaxed convergence theorem for this kind of algorithm is presented (Theorem 20). As a corollary, we get an approximation theorem for value function based reinforcement learning methods in (ε, δ)-MDPs (Corollary 21).

4. Furthermore, we illustrate our results through three classical RL algorithms. Relaxed convergence properties in (ε,δ)-MDPs for asynchronous value iteration, Q-learning and temporal difference learning are deduced. Later, we show that our approach could also be applied to investigate approximate dynamic programming methods.

5. We also present numerical experiments which highlight some features of working in varying environments. First, two simple stochastic iterative algorithms, a "well-behaving" and a "pathological" one, are shown. Regarding learning, we illustrate the effects of environmental changes through two problems: scheduling and grid world.

2. Definitions and Preliminaries

Sequential decision making under the presence of uncertainties is often modeled by MDPs (Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998; Feinberg and Shwartz, 2002). This section contains the basic definitions, the applied notations and some preliminaries.

Definition 1 *By a (finite, discrete-time, stationary, fully observable) Markov decision process (MDP) we mean a stochastic system characterized by a 6-tuple $\langle \mathbb{X}, \mathbb{A}, \mathcal{A}, p, g, \alpha \rangle$, where the components are as follows: $\mathbb{X}$ is a finite set of discrete states and $\mathbb{A}$ is a finite set of control actions. Mapping $\mathcal{A} : \mathbb{X} \to \mathcal{P}(\mathbb{A})$ is the availability function that renders a set of actions available to each state where $\mathcal{P}$ denotes the power set. The transition function is given by $p : \mathbb{X} \times \mathbb{A} \to \Delta(\mathbb{X})$, where $\Delta(\mathbb{X})$ is the set of all probability distributions over $\mathbb{X}$. Let $p(y|x,a)$ denote the probability of arrival at state y after executing action $a \in \mathcal{A}(x)$ in state x. The immediate-cost function is defined by $g : \mathbb{X} \times \mathbb{A} \to \mathbb{R}$, where $g(x,a)$ is the cost of taking action a in state x. Finally, $\alpha \in [0,1)$ denotes the discount rate.*

An interpretation of an MDP can be given, which viewpoint is often taken in RL, if we consider an agent that acts in an uncertain environment. The agent receives information about the state of the environment x, at each state x the agent is allowed to choose an action $a \in \mathcal{A}(x)$. After the action is selected, the environment moves to the next state according to the probability distribution $p(x,a)$ and the decision-maker collects its one-step cost, $g(x,a)$. The aim of the agent is to find an optimal behavior (policy), such that applying this strategy minimizes the expected cumulative costs.

Definition 2 *A (stationary, Markovian) control policy determines the action to take in each state. A deterministic policy, $\pi : \mathbb{X} \to \mathbb{A}$, is simply a function from states to control actions. A randomized policy, $\pi : \mathbb{X} \to \Delta(\mathbb{A})$, is a function from states to probability distributions over actions. We denote the probability of executing action a in state x by $\pi(x)(a)$ or, for short, by $\pi(x,a)$. Unless indicated otherwise, we consider randomized policies.*

For any $\widetilde{x}_0 \in \Delta(\mathbb{X})$ initial probability distribution of the states, the transition probabilities p together with a control policy π completely determine the progress of the system in a stochastic sense, namely, they define a *homogeneous Markov chain* on $\mathbb{X}$,

$$\widetilde{x}_{t+1} = P(\pi)\widetilde{x}_t,$$

where $\widetilde{x}_t$ is the state probability distribution vector of the system at time t and $P(\pi)$ denotes the probability transition matrix induced by control policy π,

$$[P(\pi)]_{x,y} = \sum_{a \in \mathbb{A}} p(y|x,a)\,\pi(x,a).$$

Definition 3 *The value or cost-to-go function of a policy π is a function from states to costs, $J^\pi : \mathbb{X} \to \mathbb{R}$. Function $J^\pi(x)$ gives the expected value of the cumulative (discounted) costs when the system is in state x and it follows policy π thereafter,*

$$J^\pi(x) = \mathbb{E}\left[\sum_{t=0}^{N} \alpha^t g(X_t, A_t^\pi) \,\middle|\, X_0 = x\right], \tag{2}$$

where X_t and A_t^π are random variables, A_t^π is selected according to control policy π and the distribution of X_{t+1} is $p(X_t, A_t^\pi)$. The horizon of the problem is denoted by $N \in \mathbb{N} \cup \{\infty\}$. Unless indicated otherwise, we will always assume that the horizon is infinite, $N = \infty$.

Definition 4 *We say that $\pi_1 \leq \pi_2$ if and only if $\forall x \in \mathbb{X} : J^{\pi_1}(x) \leq J^{\pi_2}(x)$. A control policy is (uniformly) optimal if it is less than or equal to all other control policies.*

There always exists at least one optimal policy (Sutton and Barto, 1998). Although there may be many optimal policies, they all share the same unique optimal cost-to-go function, denoted by J^*. This function must satisfy the Bellman optimality equation (Bertsekas and Tsitsiklis, 1996), $TJ^* = J^*$, where T is the *Bellman operator*, defined for all $x \in \mathbb{X}$, as

$$(TJ)(x) = \min_{a \in \mathcal{A}(x)} \Big[g(x,a) + \alpha \sum_{y \in \mathbb{X}} p(y \,|\, x, a) J(y) \Big].$$

Definition 5 *We say that function $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$, $\mathcal{Y}$ are normed spaces, is Lipschitz continuous if there exists a $\beta \geq 0$ such that $\forall x_1, x_2 \in \mathcal{X} : \|f(x_1) - f(x_2)\|_\mathcal{Y} \leq \beta \|x_1 - x_2\|_\mathcal{X}$, where $\|\cdot\|_\mathcal{X}$ and $\|\cdot\|_\mathcal{Y}$ denote the norm of $\mathcal{X}$ and $\mathcal{Y}$, respectively. The smallest such β is called the Lipschitz constant of f. Henceforth, assume that $\mathcal{X} = \mathcal{Y}$. If the Lipschitz constant $\beta < 1$, then the function is called a contraction. A mapping is called a pseudo-contraction if there exists an $x^* \in \mathcal{X}$ and a $\beta \geq 0$ such that $\forall x \in \mathcal{X}$, we have $\|f(x) - x^*\|_\mathcal{X} \leq \beta \|x - x^*\|_\mathcal{X}$.*

Naturally, every contraction mapping is also a pseudo-contraction, however, the opposite is not true. The pseudo-contraction condition implies that x^* is the fixed point of function f, namely, $f(x^*) = x^*$, moreover, x^* is unique, thus, f cannot have other fixed points.

It is known that the Bellman operator is a supremum norm contraction with Lipschitz constant α. In case we consider *stochastic shortest path* (SSP) problems, which arise if the MDP has an absorbing terminal (goal) state, then the Bellman operator becomes a pseudo-contraction in the weighted supremum norm (Bertsekas and Tsitsiklis, 1996).

From a given value function J, it is straightforward to get a policy, for example, by applying a *greedy* and deterministic policy (w.r.t. J), that always selects actions with minimal costs,

$$\pi(x) \in \operatorname*{arg\,min}_{a \in \mathcal{A}(x)} \Big[g(x,a) + \alpha \sum_{y \in \mathbb{X}} p(y \,|\, x, a) J(y) \Big].$$

Similarly to the definition of J^π, one can define *action-value* functions of control polices,

$$Q^\pi(x,a) = \mathbb{E}\left[\sum_{t=0}^{N} \alpha^t g(X_t, A_t^\pi) \,\middle|\, X_0 = x, A_0^\pi = a\right],$$

where the notations are the same as in (2). MDPs have an extensively studied theory and there exist a lot of exact and approximate solution methods, for example, value iteration, policy iteration, the Gauss-Seidel method, Q-learning, Q(λ), SARSA and TD(λ)—temporal difference learning (Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998; Feinberg and Shwartz, 2002). Most of these reinforcement learning algorithms work by iteratively approximating the optimal value function and typically consider stationary environments.

If J is "close" to J^*, then the greedy policy with one-stage lookahead based on J will also be "close" to an optimal policy, as it was proven by Bertsekas and Tsitsiklis (1996):

Theorem 6 *Let M be a discounted MDP and J is an arbitrary value function. The value function of the greedy policy based on J is denoted by J^π. Then, we have*

$$\|J^\pi - J^*\|_\infty \leq \frac{2\alpha}{1-\alpha} \|J - J^*\|_\infty,$$

where $\|\cdot\|_\infty$ denotes the supremum norm, namely $\|f\|_\infty = \sup\{|f(x)| : x \in domain(f)\}$. Moreover, there exists an $\varepsilon > 0$ such that if $\|J - J^*\|_\infty < \varepsilon$ then $J^* = J^\pi$.

Consequently, if we could obtain a good approximation of the optimal value function, then we immediately had a good control policy, as well, for example, the greedy policy with respect to our approximate value function. Therefore, the main question for most RL approaches is that how a good approximation of the optimal value function could be achieved.

3. Asymptotic Bounds for Generalized Value Iteration

In this section we will briefly overview a unified framework to analyze value function based reinforcement learning algorithms. We will use this approach later when we prove convergence properties in changing environments. The theory presented in this section was developed by Szepesvári and Littman (1999) and was extended by Szita et al. (2002).

3.1 Generalized Value Functions and Approximate Convergence

Throughout the paper we denote the set of value functions by $\mathcal{V}$ which contains, in general, all bounded real-valued functions over an arbitrary set $\mathcal{X}$, for example, $\mathcal{X} = \mathbb{X}$, in the case of state-value functions, or $\mathcal{X} = \mathbb{X} \times \mathbb{A}$, in the case of action-value functions. Note that the set of value functions, $\mathcal{V} = \mathcal{B}(\mathcal{X})$, where $\mathcal{B}(\mathcal{X})$ denotes the set of all bounded real-valued functions over set $\mathcal{X}$, is a normed space, for example, with the supremum norm. Naturally, bounded functions constitute no real restriction in case of analyzing finite MDPs.

Definition 7 *We say that a sequence of random variables, denoted by X_t, κ-approximates random variable X with $\kappa \geq 0$, in a given norm, if we have*

$$\mathbb{P}\left(\limsup_{t\to\infty} \|X_t - X\| \leq \kappa\right) = 1. \tag{3}$$

Hence, the "meaning" of this definition is that sequence X_t converges almost surely to an environment of X and the radius of this environment is less than or equal to a given constant κ. Naturally, this definition is weaker (more general) than the probability one convergence, which arises as a special case, when $\kappa = 0$. In the paper we will always consider convergence in the supremum norm.

3.2 Relaxed Convergence of Generalized Value Iteration

A general form of value iteration type algorithms can be given as follows,

$$V_{t+1} = H_t(V_t, V_t),$$

where H_t is a random operator on $\mathcal{V} \times \mathcal{V} \to \mathcal{V}$ (Szepesvári and Littman, 1999). Consider, for example, the SARSA (state-action-reward-state-action) algorithm which is a model-free policy evaluation method. It aims at finding Q^π for a given policy π and it is defined as

$$Q_{t+1}(x,a) = (1-\gamma_t(x,a))\,Q_t(x,a) + \gamma_t(x,a)(g(x,a) + \alpha\, Q_t(Y,B)),$$

where $\gamma_t(x,a)$ denotes the stepsize associated with state x and action a at time t; Y and B are random variables, Y is generated from the pair (x,a) by simulation, that is, according to the distribution $p(x,a)$, and the distribution of B is $\pi(Y)$. In this case, H_t is defined as

$$H_t(Q_a, Q_b)(x,a) = (1-\gamma_t(x,a))\,Q_a(x,a) + \gamma_t(x,a)(g(x,a) + \alpha\, Q_b(Y,B)), \tag{4}$$

for all x and a. Therefore, the SARSA algorithm takes the form $Q_{t+1} = H_t(Q_t, Q_t)$.

Definition 8 *We say that the operator sequence H_t κ-approximates operator $H : \mathcal{V} \to \mathcal{V}$ at $V \in \mathcal{V}$ if for any initial $V_0 \in \mathcal{V}$ the sequence $V_{t+1} = H_t(V_t, V)$ κ-approximates HV.*

The next theorem (Szita et al., 2002) will be an important tool for proving convergence results for value function based RL algorithms in varying environments.

Theorem 9 *Let H be an arbitrary mapping with fixed point V^*, and let H_t κ-approximate H at V^* over set $\mathcal{X}$. Additionally, assume that there exist random functions $0 \le F_t(x) \le 1$ and $0 \le G_t(x) \le 1$ satisfying the four conditions below with probability one*

1. *For all $V_1, V_2 \in \mathcal{V}$ and for all $x \in \mathcal{X}$,*

$$|H_t(V_1, V^*)(x) - H_t(V_2, V^*)(x)| \le G_t(x)\,|V_1(x) - V_2(x)|\,.$$

2. *For all $V_1, V_2 \in \mathcal{V}$ and for all $x \in \mathcal{X}$,*

$$|H_t(V_1, V^*)(x) - H_t(V_1, V_2)(x)| \le F_t(x)\,\|V^* - V_2\|_\infty\,.$$

3. *For all $k > 0$, $\prod_{t=k}^{n} G_t(x)$ converges to zero uniformly in x as n increases.*

4. *There exist $0 \le \xi < 1$ such that for all $x \in \mathcal{X}$ and sufficiently large t,*

$$F_t(x) \le \xi\,(1 - G_t(x)).$$

Then, $V_{t+1} = H_t(V_t, V_t)$ κ'-approximates V^* over $\mathcal{X}$ for any $V_0 \in \mathcal{V}$, where $\kappa' = 2\kappa/(1-\xi)$.

Usually, functions F_t and G_t can be interpreted as the ratio of mixing the two arguments of operator H_t. In the case of the SARSA algorithm, described above by (4), $\mathcal{X} = \mathbb{X} \times \mathbb{A}$, $G_t(x,a) = (1-\gamma_t(x,a))$ and $F_t(x,a) = \alpha\gamma_t(x,a)$ would be a suitable choice.

One of the most important aspects of this theorem is that it shows how to reduce the problem of approximating V^* with $V_t = H_t(V_t, V_t)$ type operators to the problem of approximating it with a $V_t' = H_t(V_t', V^*)$ sequence, which is, in many cases, much easier to be dealt with. This makes, for example, the convergence of Watkins' Q-learning a consequence of the classical Robbins-Monro theory (Szepesvári and Littman, 1999; Szita et al., 2002).

4. Value Function Bounds for Environmental Changes

In many control problems it is typically not possible to "practise" in the real environment, only a dynamic *model* is available to the system and this model can be used for predicting how the environment will respond to the control signals (model predictive control). MDP based solutions usually work by *simulating* the environment with the model, through simulation they produce *simulated experience* and by *learning* from these experience they improve their value functions. Computing an approximately optimal value function is essential because, as we have seen (Theorem 6), close approximations to optimal value functions lead directly to good policies. Though, there are alternative approaches which directly approximate optimal control policies (see Sutton et al., 2000). However, what happens if the model was inaccurate or the environment had changed slightly? In what follows we investigate the effects of environmental changes on the optimal value function. For continuous Markov processes questions like these were already analyzed (Gordienko and Salem, 2000; Favero and Runggaldier, 2002; de Oca et al., 2003), hence, we will focus on *finite* MDPs.

The theorems of this section have some similarities with two previous results. First, Munos and Moore (2000) studied the dependence of the Bellman operator on the transition-probabilities and the immediate-costs. Later, Kearns and Singh (2002) applied a *simulation lemma* to deduce polynomial time bounds to achieve near-optimal return in MDPs. This lemma states that if two MDPs differ only in their transition and cost functions and we want to approximate the value function of a fixed policy concerning one of the MDPs in the other MDP, then how close should we choose the transitions and the costs to the original MDP relative to the mixing time or the horizon time.

4.1 Changes in the Transition-Probability Function

First, we will see that the optimal value function of a *discounted* MDP Lipschitz continuously depends on the transition-probability function. This question was analyzed by Müller (1996), as well, but the presented version of Theorem 10 was proven by Kalmár et al. (1998).

Theorem 10 *Assume that two discounted MDPs differ only in their transition functions, denoted by p_1 and p_2. Let the corresponding optimal value functions be J_1^* and J_2^*, then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{n\alpha\|g\|_\infty}{(1-\alpha)^2}\|p_1 - p_2\|_\infty,$$

recall that n is the size of the state space and $\alpha \in [0,1)$ is the discount rate.

A disadvantage of this theorem is that the estimation heavily depends on the size of the state space, *n*. However, this bound can be improved if we consider an induced matrix norm for transition-probabilities instead of the supremum norm. The following theorem presents our improved estimation for transition changes. Its proof can be found in the appendix.

Theorem 11 *With the assumptions and notations of Theorem 10, we have*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{\alpha\|g\|_\infty}{(1-\alpha)^2}\|p_1 - p_2\|_1,$$

where $\|\cdot\|_1$ is a norm on $f : \mathbb{X}\times\mathbb{A}\times\mathbb{X} \to \mathbb{R}$ type functions, for example, $f(x,a,y) = p(y|x,a)$,

$$\|f\|_1 = \max_{x,a}\sum_{y\in\mathbb{X}} |f(x,a,y)|. \tag{5}$$

If we consider f as a matrix which has a column for each state-action pair $(x,a) \in \mathbb{X} \times \mathbb{A}$ and a row for each state $y \in \mathbb{X}$, then the above definition gives us the usual "maximum absolute column sum norm" definition for matrices, which is conventionally denoted by $\|\cdot\|_1$.

It is easy to see that for all f, we have $\|f\|_1 \leq n\|f\|_\infty$, where n is size of the state space. Therefore, the estimation of Theorem 11 is at least as good as the estimation of Theorem 10. In order to see that it is a real improvement consider, for example, the case when we choose a particular state-action pair, $(\hat{x},\hat{a})$, and take a p_1 and p_2 that only differ in $(\hat{x},\hat{a})$. For example, $p_1(\hat{x},\hat{a}) = \langle 1,0,0,\ldots,0\rangle$ and $p_2(\hat{x},\hat{a}) = \langle 0,1,0,\ldots,0\rangle$, and they are equal for all other $(x,a) \neq (\hat{x},\hat{a})$. Then, by definition, $\|p_1 - p_2\|_1 = 2$, but $n\|p_1 - p_2\|_\infty = n$. Consequently, in this case, we have improved the bound of Theorem 10 by a factor of $2/n$.

4.2 Changes in the Immediate-Cost Function

The same kind of Lipschitz continuity can be proven in case of changes in the cost function.

Theorem 12 *Assume that two discounted MDPs differ only in the immediate-costs functions, g_1 and g_2. Let the corresponding optimal value functions be J_1^* and J_2^*, then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{1}{1-\alpha}\|g_1 - g_2\|_\infty .$$

4.3 Changes in the Discount Factor

The following theorem shows that the change of the value function can also be estimated in case there were changes in the discount rate (all proofs can be found in the appendix).

Theorem 13 *Assume that two discounted MDPs differ only in the discount factors, denoted by $\alpha_1, \alpha_2 \in [0,1)$. Let the corresponding optimal value functions be J_1^* and J_2^*, then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{|\alpha_1 - \alpha_2|}{(1-\alpha_1)(1-\alpha_2)}\|g\|_\infty .$$

The next example demonstrates, however, that this dependence is not Lipschitz continuous. Consider, for example, an MDP that has only one state x and one action a. Taking action a loops back deterministically to state x with cost $g(x,a) = 1$. Suppose that the MDP has discount factor $\alpha_1 = 0$, thus, $J_1^*(x) = 1$. Now, if we change the discount rate to $\alpha_2 \in (0,1)$, then $|\alpha_1 - \alpha_2| < 1$ but $\|J_1^* - J_2^*\|_\infty$ could be arbitrarily large, since $J_2^*(x) \to \infty$ as $\alpha_2 \to 1$.

At the same time, we can notice that if we fix a constant $\alpha_0 < 1$ and only allow discount factors from the interval $[0,\alpha_0]$, then this dependence became Lipschitz continuous, as well.

4.4 Case of Action-Value Functions

Many reinforcement learning algorithms, such as Q-learning, work with action-value functions which are important, for example, for model-free approaches. Now, we investigate how the previously presented theorems apply to this type of value functions. The optimal action-value function, denoted by Q^*, is defined for all state-action pair (x,a) by

$$Q^*(x,a) = g(x,a) + \alpha \sum_{y \in \mathbb{X}} p(y|x,a)J^*(y),$$

where J^* is the optimal state-value function. Note that in the case of the optimal action-value function, first, we take a given action (which can have very high cost) and, only after that the action was taken, follow an optimal policy. Thus, we can estimate $\|Q^*\|_\infty$ by

$$\|Q^*\|_\infty \leq \|g\|_\infty + \alpha \|J^*\|_\infty .$$

Nevertheless, the next lemma shows that the same estimations can be derived for environmental changes in the case of action-value functions as in the case of state-value functions.

Lemma 14 *Assume that we have two discounted MDPs which differ only in the transition-probability functions or only in the immediate-cost functions or only in the discount factors. Let the corresponding optimal action-value functions be Q_1^* and Q_2^*, respectively. Then, the bounds for $\|J_1^* - J_2^*\|_\infty$ of Theorems 11, 12 and 13 are also bounds for $\|Q_1^* - Q_2^*\|_\infty$.*

4.5 Further Remarks on Inaccurate Models

In this section we saw that the optimal value function of a discounted MDP depends smoothly on the transition function, the cost function and the discount rate. This dependence is of Lipschitz type in the first two cases and non-Lipschitz for discount rates.

If we treat one of the MDPs in the previous theorems as a system which describes the "real" behavior of the environment and the other MDP as our model, then these results show that even if the model is slightly inaccurate or there were changes in the environment, the optimal value function based on the model cannot be arbitrarily wrong from the viewpoint of the environment. These theorems are of special interest because in "real world" problems the transition-probabilities and the immediate-costs are mostly estimated only, for example, by statistical methods from historical data. Later, we will see that changes in the discount rate can be traced back to changes in the cost function (Lemma 18), therefore, it is sufficient to consider transition and cost changes. The following corollary summarizes the results.

Corollary 15 *Assume that two discounted MDPs (E and M) differ only in their transition functions and their cost functions. Let the corresponding transition and cost functions be denoted by p_E, p_M and g_E, g_M, respectively. The corresponding optimal value functions are denoted by J_E^* and J_M^*. The value function in E of the deterministic and greedy policy (π) with one stage-lookahead that is based upon J_M^* is denoted by J_E^π. Then,*

$$\|J_E^\pi - J_E^*\|_\infty \leq \frac{2\alpha}{1-\alpha}\left[\frac{\|g_E - g_M\|_\infty}{1-\alpha} + \frac{c\,\alpha\,\|p_E - p_M\|_1}{(1-\alpha)^2}\right],$$

where $c = \min\{\|g_E\|_\infty, \|g_M\|_\infty\}$ and $\alpha \in [0,1)$ is the discount factor.

The proof simply follows from Theorems 6, 11 and 12 and from the triangle inequality. Another interesting question is the effects of environmental changes on the value function of a *fixed* control policy. However, it is straightforward to prove (Csáji, 2008) that the same estimations can be derived for $\|J_1^\pi - J_2^\pi\|_\infty$, where π is an arbitrary (stationary, Markovian, randomized) control policy, as the estimations of Theorems 10, 11, 12 and 13.

Note that the presented theorems are only valid in case of *discounted* MDPs. Though, a large part of the MDP related research studies the expected total discounted cost optimality criterion, in

some cases discounting is inappropriate and, therefore, there are alternative optimality approaches, as well. A popular alternative approach is to optimize the *expected average cost* (Feinberg and Shwartz, 2002). In this case the value function is defined as

$$J^\pi(x) = \limsup_{N\to\infty} \frac{1}{N}\,\mathbb{E}\left[\sum_{t=0}^{N-1} \alpha^t g(X_t, A_t^\pi) \;\middle|\; X_0 = x\right],$$

where the notations are the same as previously, for example, as applied in Equation (2).

Regarding the validity of the results of Section 4 concerning MDPs with the expected average cost minimization objective, we can recall that, in the case of finite MDPs, discounted cost offers a good approximation to the other optimality criterion. More precisely, it can be shown that there exists a large enough $\alpha_0 < 1$ such that $\forall\,\alpha \in (\alpha_0, 1)$ optimal control policies for the discounted cost problem are also optimal for the average cost problem (Feinberg and Shwartz, 2002). These policies are called *Blackwell optimal*.

5. Learning in Varying Environments

In this section we investigate how value function based learning methods can act in environments which may change over time. However, without restrictions, this approach would be too general to establish convergence results. Therefore, we restrict ourselves to the case when the changes remain bounded over time. In order to precisely define this concept, the idea of (ε, δ)-MDPs is introduced, which is a generalization of classical MDPs and ε-MDPs. First, we recall the definition of ε-MDPs (Kalmár et al., 1998; Szita et al., 2002).

Definition 16 *A sequence of MDPs $(\mathcal{M}_t)_{t=1}^\infty$ is called an ε-MDP with $\varepsilon > 0$ if the MDPs differ only in their transition-probability functions, denoted by p_t for $\mathcal{M}_t$, and there exists an MDP with transition function p, called the base MDP, such that $\sup_t \|p - p_t\| \leq \varepsilon$.*

5.1 Varying Environments: (ε, δ)-MDPs

Now, we extend the idea described above. The following definition of (ε, δ)-MDPs generalizes the concept of ε-MDPs in two ways. First, we also allow the cost function to change over time and, additionally, we require the changes to remain bounded by parameters ε and δ only asymptotically, in the limit. A finite number of large deviations is tolerated.

Definition 17 *A tuple $\langle \mathbb{X}, \mathbb{A}, \mathcal{A}, \{p_t\}_{t=1}^\infty, \{g_t\}_{t=1}^\infty, \alpha\rangle$ is an (ε, δ)-MDP with $\varepsilon, \delta \geq 0$, if there exists an MDP $\langle \mathbb{X}, \mathbb{A}, \mathcal{A}, p, g, \alpha\rangle$, called the base MDP, such that*

1. $\limsup_{t\to\infty} \|p - p_t\| \leq \varepsilon$
2. $\limsup_{t\to\infty} \|g - g_t\| \leq \delta$

The optimal value function of the base MDP and of the current MDP at time t (which MDP has transition function p_t and cost function g_t) are denoted by J^* and J_t^*, respectively.

In order to keep the analysis as simple as possible, we do not allow the discount rate parameter α to change over time; not only because, for example, with Theorem 13 at hand, it would be straightforward to extend the results to the case of changing discount factors, but even more because, as

Lemma 18 demonstrates, the effects of changes in the discount rate can be incorporated into the immediate-cost function, which is allowed to change in (ε,δ)-MDPs.

Lemma 18 *Assume that two discounted MDPs, $\mathcal{M}_1$ and $\mathcal{M}_2$, differ only in the discount factors, denoted by α_1 and α_2. Then, there exists an MDP, denoted by $\mathcal{M}_3$, such that it differs only in the immediate-cost function from $\mathcal{M}_1$, thus its discount factor is α_1, and it has the same optimal value function as $\mathcal{M}_2$. The immediate-cost function of $\mathcal{M}_3$ is*

$$\widehat{g}(x,a) = g(x,a) + (\alpha_2 - \alpha_1) \sum_{y\in\mathbb{X}} p(y|x,a) J_2^*(y),$$

where p is the probability-transition function of $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$; g is the immediate-cost function of $\mathcal{M}_1$ and $\mathcal{M}_2$; and $J_2^*(y)$ denotes the optimal cost-to-go function of $\mathcal{M}_2$.

On the other hand, we can notice that changes in the cost function cannot be traced back to changes in the transition function. Consider, for example, an MDP with a constant zero cost function. Then, no matter what the transition-probabilities are, the optimal value function remains zero. However, we may achieve non-zero optimal value function values if we change the immediate-cost function. Therefore, (ε,δ)-MDPs cannot be traced back to ε-MDPs.

Now, we briefly investigate the applicability of (ε,δ)-MDPs and a possible motivation behind them. When we model a "real world" problem as an MDP, then we typically take only the *major characteristics* of the system into account, but there could be many *hidden parameters*, as well, which may affect the transition-probabilities and the immediate-costs, however, which are not explicitly included in the model. For example, if we model a production control system as an MDP (Csáji and Monostori, 2006), then the workers' fatigue, mood or the quality of the materials may affect the durations of the tasks, but these characteristics are usually not included in the model. Additionally, the values of these hidden parameters may change over time. In these cases, we could either try to incorporate as many aspects of the system as possible into the model, which would most likely lead to *computationally intractable* results, or we could model the system as an (ε,δ)-MDP, which would result in a simplified model and, presumably, in a more tractable system.

5.2 Relaxed Convergence of Stochastic Iterative Algorithms

In this section we present a general relaxed convergence theorem for a large class of stochastic iterative algorithms. Later, we will apply this theorem to investigate the convergence properties of value function based reinforcement learning methods in (ε,δ)-MDPs.

Many learning and optimization methods can be written in a general form as a *stochastic iterative algorithm* (Bertsekas and Tsitsiklis, 1996). More precisely, as

$$V_{t+1}(x) = (1-\gamma_t(x))V_t(x) + \gamma_t(x)((K_tV_t)(x) + W_t(x)), \tag{6}$$

where $V_t \in \mathcal{V}$, operator $K_t : \mathcal{V} \to \mathcal{V}$ acts on value functions, each $\gamma_t(x)$ is a random variable which determines the stepsize and $W_t(x)$ is also a random variable, a noise parameter.

Regarding reinforcement learning algorithms, for example, (asynchronous) value iteration, Gauss-Seidel methods, Q-learning and TD(λ) – temporal difference learning can be formulated this way. We will show that under suitable conditions these algorithms work in (ε,δ)-MDPs, more precisely, κ-approximation to the optimal value function of the base MDP will be proven.

Now, in order to provide our relaxed convergence result, we introduce assumptions on the noise parameters, on the stepsize parameters and on the value function operators.

Definition 19 *We denote the history of the algorithm until time t by $\mathcal{F}_t$, defined as*

$$\mathcal{F}_t = \{V_0, \ldots, V_t, W_0, \ldots, W_{t-1}, \gamma_0, \ldots, \gamma_t\}.$$

The sequence $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \mathcal{F}_2 \subseteq \ldots$ can be seen as a *filtration*, viz., as an increasing sequence of σ-fields. The set $\mathcal{F}_t$ represents the information available at each time t.

Assumption 1 *There exits a constant $C > 0$ such that for all state x and time t, we have*

$$\mathbb{E}[W_t(x) \,|\, \mathcal{F}_t] = 0 \quad \textit{and} \quad \mathbb{E}\left[W_t^2(x) \,|\, \mathcal{F}_t\right] < C < \infty.$$

Regarding the stepsize parameters, γ_t, we make the "usual" stochastic approximation assumptions. Note that there is a separate stepsize parameter for each possible state.

Assumption 2 *For all x and t, $0 \leq \gamma_t(x) \leq 1$, and we have with probability one*

$$\sum_{t=0}^{\infty} \gamma_t(x) = \infty \quad \textit{and} \quad \sum_{t=0}^{\infty} \gamma_t^2(x) < \infty.$$

Intuitively, the first requirement guarantees that the stepsizes are able to overcome the effects of finite noises, while the second criterion ensures that they eventually converge.

Assumption 3 *For all t, operator $K_t : \mathcal{V} \to \mathcal{V}$ is a supremum norm contraction mapping with Lipschitz constant $\beta_t < 1$ and with fixed point V_t^*. Formally, for all $V_1, V_2 \in \mathcal{V}$,*

$$\|K_t V_1 - K_t V_2\|_\infty \leq \beta_t \|V_1 - V_2\|_\infty.$$

Let us introduce a common Lipschitz constant $\beta_0 = \limsup_{t\to\infty} \beta_t$, and assume that $\beta_0 < 1$.

Because our aim is to analyze changing environments, each K_t operator can have different fixed points and different Lipschitz constants. However, to avoid the progress of the algorithm to "slow down" infinitely, we should require that $\limsup_{t\to\infty} \beta_t < 1$. In the next section, when we apply this theory to the case of (ε, δ)-MDPs, each value function operator can depend on the current MDP at time t and, thus, can have different fixed points.

Now, we present a theorem (its proof can be found in the appendix) that shows how the function sequence generated by iteration (6) can converge to an environment of a function.

Theorem 20 *Suppose that Assumptions 1-3 hold and let V_t be the sequence generated by iteration (6). Then, for any $V^*, V_0 \in \mathcal{V}$, the sequence V_t κ-approximates function V^* with*

$$\kappa = \frac{4\rho}{1-\beta_0} \quad \textit{where} \quad \rho = \limsup_{t\to\infty} \|V_t^* - V^*\|_\infty.$$

This theorem is very general, it is valid even in the case of non-finite MDPs. Notice that V^* can be an *arbitrary* function but, naturally, the radius of the environment of V^*, which the sequence V_t almost surely converges to, depends on $\limsup_{t\to\infty} \|V_t^* - V^*\|_\infty$.

If we take a closer look at the proof, we can notice that the theorem is still valid if each K_t is only a *pseudo-contraction* but, additionally, it also attracts points to V^*. Formally, it is enough if we assume that for all $V \in \mathcal{V}$, we have $\|K_t V - K_t V_t^*\|_\infty \leq \beta_t \|V - V_t^*\|_\infty$ and $\|K_t V - K_t V^*\|_\infty \leq \beta_t \|V - V^*\|_\infty$ for a suitable $\beta_t < 1$. This remark could be important in case we want to apply Theorem 20 to changing *stochastic shortest path (SSP)* problems.

5.2.1 A SIMPLE NUMERICAL EXAMPLE

Consider a one dimensional stochastic process characterized by the iteration

$$v_{t+1} = (1-\gamma_t)v_t + \gamma_t(K_t(v_t) + w_t), \tag{7}$$

where γ_t is the learning rate and w_t is a noise term. Let us suppose we have n alternating operators k_i with Lipschitz constants $b_i < 1$ and fixed points v_i^* where $i \in \{0,\ldots,n-1\}$,

$$k_i(v) = v + (1-b_i)(v_i^* - v).$$

The current operator at time t is $K_t = k_i$ (thus, $V_t^* = v_i^*$ and $\beta_t = b_i$) if $i \equiv t \pmod n$, that is, if i is congruent with t modulo n: if they have the same remainder when they are divided by n. In other words, we apply a *round-robin* type schedule for the operators.

Figure 1 shows that the trajectories remained close to the fixed points. The figure illustrates the case of two (-1 and 1) and six ($-3,-2,-1,1,2,3$) alternating fixed points.

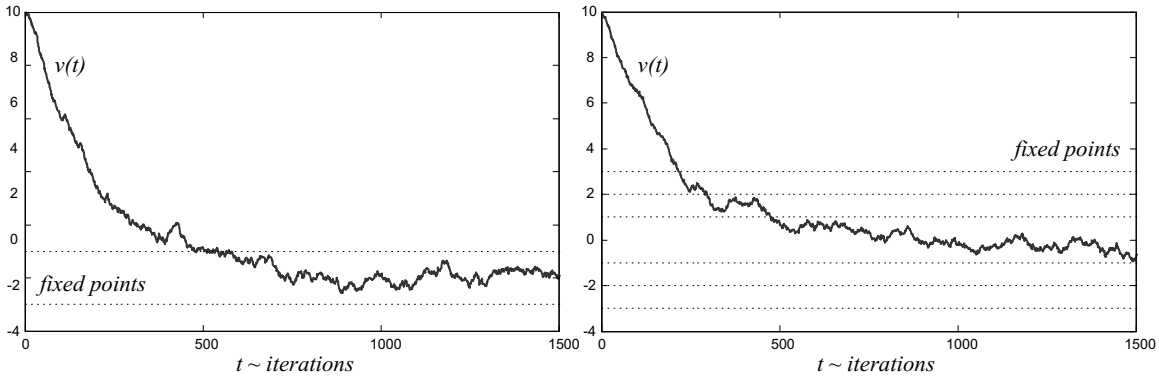

Figure 1: Trajectories generated by (7) with two (left) and six (right) fixed points.

5.2.2 A PATHOLOGICAL EXAMPLE

During this example we will restrict ourselves to deterministic functions. According to the *Banach fixed point theorem*, if we have a contraction mapping f over a complete metric space with fixed point $v^* = f(v^*)$, then, for any initial v_0 the sequence $v_{t+1} = f(v_t)$ converges to v^*. It could be thought that this result can be easily generalized to the case of alternating operators. For example, suppose we have n alternating contraction mappings k_i with Lipschitz constants $b_i < 1$ and fixed points v_i^*, respectively, where $i \in \{0,\ldots,n-1\}$, and we apply them iteratively starting from an arbitrary v_0, viz., $v_{t+1} = K_t(v_t)$, where $K_t = k_i$ if $i \equiv t \pmod n$. One may think that since each k_i attracts the point towards its fixed point, the sequence v_t converges to the *convex hull* of the fixed points. However, as the following example demonstrates, this is not the case, since it is possible that the point moves away from the convex hull and, in fact, it gets farther and farther after each iteration.

Now, let us consider two one-dimensional functions, $k_i : \mathbb{R} \to \mathbb{R}$, where $i \in \{a,b\}$, defined below by Equation (8). It can be easily proven that these functions are contractions with fixed points v_i^*

and Lipschitz constants b_i (in Figure 2, $v_a^* = 1$, $v_b^* = -1$ and $b_i = 0.9$).

$$k_i(v) = \begin{cases} v + (1-b_i)(v_i^* - v) & \text{if } sgn(v_i^*) = sgn(v - v_i^*), \\ v_i^* + (v_i^* - v) + (1-b_i)(v - v_i^*) & \text{otherwise,} \end{cases} \tag{8}$$

where $sgn(\cdot)$ denotes the signum[1] function. Figure 2 demonstrates that even if the iteration starts from the middle of the convex hull (from the center of mass), $v_0 = 0$, it starts getting farther and farther from the fixed points in each step when we apply k_a and k_b after each other. Nevertheless,

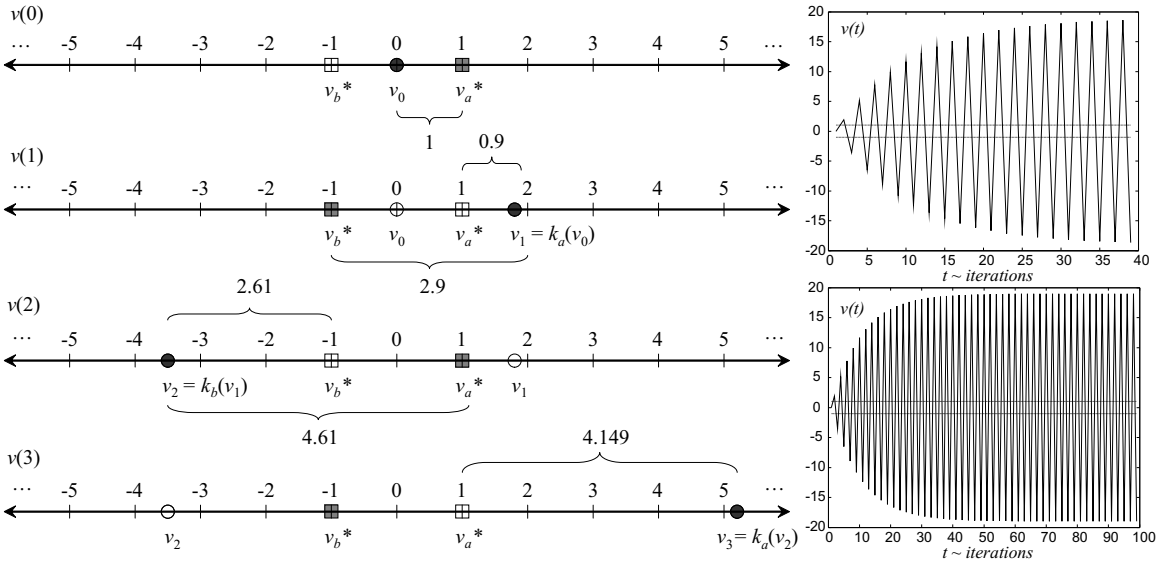

Figure 2: A deterministic pathological example, generated by the iterative application of (8). The left part demonstrates the first steps, while the two images on the right-hand side show the behavior of the trajectory in the long run.

the following argument shows that sequence v_t cannot get arbitrarily far from the fixed points. Let us denote the *diameter* of the convex hull of the fixed points by ρ. Since this convex hull is a polygon (where the vertices are fixed points) $\rho = \max_{i,j} \|v_i^* - v_j^*\|$. Furthermore, let β_0 be defined as $\beta_0 = \max_i b_i$ and d_t as $d_t = \min_i \|v_i^* - v_t\|$. Then, it can be proven that for all t, we have $d_{t+1} \leq \beta_0(2\rho + d_t)$. If we assume that $d_{t+1} \geq d_t$, then it follows that $d_t \leq d_{t+1} \leq \beta_0(2\rho + d_t)$. After rearrangement, we get the following inequality

$$d_t \leq \frac{2\beta_0\rho}{1-\beta_0} = \phi(\beta_0, \rho).$$

Therefore, $d_t > \phi(\beta_0, \rho)$ implies that $d_{t+1} < d_t$. Consequently, if v_t somehow got farther than $\phi(\beta_0, \rho)$, in the next step it would inevitably be attracted towards the fixed points. It is easy to see that this argument is valid in an arbitrary normed space, as well.

1. $sgn(x) = 0$ if $x = 0$, $sgn(x) = -1$ if $x < 0$ and $sgn(x) = 1$ if $x > 0$.

5.3 Reinforcement Learning in (ε,δ)-MDPs

In case of finite (ε,δ)-MDPs we can formulate a relaxed convergence theorem for value function based reinforcement learning algorithms, as a corollary of Theorem 20. Suppose that $\mathcal{V}$ consists of state-value functions, namely, $\mathcal{X}=\mathbb{X}$. Then, we have

$$\limsup_{t\to\infty}\|J^*-J_t^*\|_\infty \le d(\varepsilon,\delta),$$

where J_t^* is the optimal value function of the MDP at time t and J^* is the optimal value function of the base MDP. In order to calculate $d(\varepsilon,\delta)$, Theorems 11 (or 10), 12 and the triangle inequality could be applied. Assume, for example, that we use the supremum norm, $\|\cdot\|_\infty$, for cost functions and $\|\cdot\|_1$, defined by Equation (5), for transition functions. Then,

$$d(\varepsilon,\delta)=\frac{\varepsilon\alpha\|g\|_\infty}{(1-\alpha)^2}+\frac{\delta}{1-\alpha},$$

where g is the cost function of the base MDP. Now, by applying Theorem 20, we have

Corollary 21 *Suppose that we have an (ε,δ)-MDP and Assumptions 1-3 hold. Let V_t be the sequence generated by iteration (6). Furthermore, assume that the fixed point of each operator K_t is J_t^*. Then, for any initial $V_0\in\mathcal{V}$, the sequence V_t κ-approximates J^* with*

$$\kappa=\frac{4d(\varepsilon,\delta)}{1-\beta_0}.$$

Notice that as parameters ε and δ go to zero, we get back to a classical convergence theorem for this kind of stochastic iterative algorithm (still in a little bit generalized form, since β_t might still change over time). Now, with the help of these results, we will investigate the convergence of some classical reinforcement learning algorithms in (ε,δ)-MDPs.

5.3.1 ASYNCHRONOUS VALUE ITERATION IN (ε,δ)-MDPS

The method of value iteration is one of the simplest reinforcement learning algorithms. In ordinary MDPs it is defined by the iteration $J_{t+1}=TJ_t$, where T is the Bellman operator. It is known that the sequence J_t converges in the supremum norm to J^* for any initial J_0 (Bertsekas and Tsitsiklis, 1996). The asynchronous variant of value iteration arises when the states are updated asynchronously, for example, only one state in each iteration. In the case of (ε,δ)-MDPs a small stepsize variant of asynchronous value iteration can be defined as

$$J_{t+1}(x)=(1-\gamma_t(x))J_t(x)+\gamma_t(x)(T_tJ_t)(x),$$

where T_t is the Bellman operator of the current MDP at time t. Since there is no noise term in the iteration, Assumption 1 is trivially satisfied. Assumption 3 follows from the fact that each T_t operator is an α contraction where α is the discount factor. Therefore, if the stepsizes satisfy Assumption 2 then, by applying Corollary 21, we have that the sequence J_t κ-approximates J^* for any initial value function J_0 with $\kappa=(4d(\varepsilon,\delta))/(1-\alpha)$.

5.3.2 Q-LEARNING IN (ε,δ)-MDPS

Watkins' Q-learning is a very popular off-policy model-free reinforcement learning algorithm (Even-Dar and Mansour, 2003). Its generalized version in ε-MDPs was studied by Szita et al. (2002). The Q-learning algorithm works with action-value functions, therefore, $\mathcal{X} = \mathbb{X} \times \mathbb{A}$, and the one-step Q-learning rule in (ε,δ)-MDPs can be defined as follows

$$Q_{t+1}(x,a) = (1-\gamma_t(x,a))Q_t(x,a) + \gamma_t(x,a)(\widetilde{T}_t Q_t)(x,a), \tag{9}$$

$$(\widetilde{T}_t Q_t)(x,a) = g_t(x,a) + \alpha \min_{B\in\mathcal{A}(Y)} Q_t(Y,B),$$

where g_t is the immediate-cost function of the current MDP at time t and Y is a random variable generated from the pair (x,a) by simulation, that is, according to the probability distribution $p_t(x,a)$, where p_t is the transition function of the current MDP at time t.

Operator $\widetilde{T}_t$ is randomized, but as it was shown by Bertsekas and Tsitsiklis (1996) in their convergence theorem for Q-learning, it can be rewritten in a form as follows

$$(\widetilde{T}_t Q)(x,a) = (\widetilde{K}_t Q)(x,a) + \widetilde{W_t}(x,a),$$

where $\widetilde{W_t}(x,a)$ is a noise term with zero mean and finite variance, and $\widetilde{K}_t$ is defined as

$$(\widetilde{K}_t Q)(x,a) = g_t(x,a) + \alpha \sum_{y\in\mathbb{X}} p_t(y \mid x,a) \min_{b\in\mathcal{A}(y)} Q(y,b).$$

Let us denote the optimal action-value function of the current MDP at time t and the base MDP by Q_t^* and Q^*, respectively. By using the fact that $J^*(x) = \min_a Q^*(x,a)$, it is easy to see that for all t, Q_t^* is the fixed point of operator $\widetilde{K}_t$ and, moreover, each $\widetilde{K}_t$ is an α contraction. Therefore, if the stepsizes satisfy Assumption 2, then the Q_t sequence generated by iteration (9) κ-approximates Q^* for any initial Q_0 with $\kappa = (4d(\varepsilon,\delta))/(1-\alpha)$.

In some situations the immediate costs are randomized, however, even in this case the relaxed convergence of Q-learning would follow as long as the random immediate costs had finite expected value and variance, which is required for satisfying Assumption 1.

5.3.3 TEMPORAL DIFFERENCE LEARNING IN (ε,δ)-MDPS

Temporal difference learning, or for short TD-learning, is a policy evaluation algorithm. It aims at finding the corresponding value function J^π for a given control policy π (Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998). It can also be used for approximating the optimal value function, for example, if we apply it together with the policy iteration algorithm.

First, we briefly review the off-line first-visit variant of TD(λ) in case of ordinary MDPs. It can be shown that the value function of a policy π can be rewritten in a form as

$$J^\pi(x) = \mathbb{E}\left[\sum_{m=0}^{\infty} (\alpha\lambda)^m D^\pi_{\alpha,m} \,\middle|\, X_0 = x\right] + J^\pi(x),$$

where $\lambda \in [0,1)$ and $D^\pi_{\alpha,m}$ denotes the "temporal difference" coefficient at time m,

$$D^\pi_{\alpha,m} = g(X_m, A^\pi_m) + \alpha J^\pi(X_{m+1}) - J^\pi(X_m),$$

where X_m, X_{m+1} and A_m^π are random variables, X_{m+1} has $p(X_m, A_m^\pi)$ distribution and A_m^π is a random variable for actions, it is selected according to the distribution $\pi(X_m)$.

Based on this observation, we can define a stochastic approximation algorithm as follows. Let us suppose that we have a generative model of the environment, for example, we can perform simulations in it. Each simulation produces a state-action-reward trajectory. We can assume that all simulations eventually end, for example, there is an absorbing termination state or we can stop the simulation after a given number of steps. Note that even in this case we can treat each trajectory as infinitely long, viz., we can define all costs after the termination as zero. The off-line first-visit TD(λ) algorithm updates the value function after each simulation,

$$J_{t+1}(x_k^t) = J_t(x_k^t) + \gamma_t(x_k^t) \sum_{m=k}^{\infty} (\alpha\lambda)^{m-k} d_{\alpha,m,t}, \tag{10}$$

where x_k^t is the state at step k in trajectory t and $d_{\alpha,m,t}$ is the temporal difference coefficient,

$$d_{\alpha,m,t} = g(x_m^t, a_m^t) + \alpha J_t(x_{m+1}^t) - J_t(x_m^t).$$

For the case of ordinary MDPs it is known that TD(λ) converges almost surely to J^π for any initial J_0 provided that each state is visited by infinitely many trajectories and the stepsizes satisfy Assumption 2. The proof is based on the observation that iteration (10) can be seen as a Robbins-Monro type stochastic iterative algorithm for finding the fixed point of $J^\pi = HJ^\pi$, where H is a contraction mapping with Lipschitz constant α (Bertsekas and Tsitsiklis, 1996). The only difference in the case of (ε, δ)-MDPs is that the environment may change over time and, therefore, operator H becomes time-dependent. However, each H_t is still an α contraction, but they potentially have different fixed points. Therefore, we can apply Theorem 20 to achieve a relaxed convergence result for off-line first-visit TD(λ) in changing environments under the same conditions as in the case of ordinary MDPs.

The convergence of the on-line every-visit variant can be proven in the same way as in the case of ordinary MDPs, viz., by showing that the difference between the two variants is of second order in the size of γ_t and hence inconsequential as γ_t diminishes to zero.

5.3.4 APPROXIMATE DYNAMIC PROGRAMMING

Most RL algorithms in their standard forms, for example, with lookup table representations, are highly intractable in practice. This phenomenon, which was named "curse of dimensionality" by Bellman, has motivated approximate approaches that result in more tractable methods, but often yield suboptimal solutions. These techniques are usually referred to as *approximate dynamic programming* (ADP). Many ADP methods are combined with simulation, but their key issue is to approximate the value function with a suitable *approximation architecture*: $V \approx \Phi(r)$, where r is a parameter vector. Direct ADP methods collect samples by using simulation, and fit the architecture to the samples. Indirect methods obtain parameter r by using an approximate version of the Bellman equation (Bertsekas, 2007).

The *power of the approximation architecture* is the smallest error that can be achieved, $\eta = \inf_r \|V^* - \Phi(r)\|$, where V^* is the optimal value function. Suppose that $\eta > 0$, then no algorithm can provide a result whose distance from V^* is less than η. Hence, the maximum that we can hope for is to converge to an environment of V^* (Bertsekas and Tsitsiklis, 1996). In what follows, we briefly investigate the connection of our results with ADP.

In general, many direct and indirect ADP methods can be formulated as follows

$$\Phi(r_{t+1}) = \Pi\big((1-\gamma_t)\Phi(r_t) + \gamma_t(B_t(\Phi(r_t)) + W_t)\big), \tag{11}$$

where $r_t \in \Theta$ is an approximation parameter, Θ is the parameter space, for example, $\Theta \subseteq \mathbb{R}^p$, $\Phi : \Theta \to \mathcal{F}$ is an approximation architecture where $\mathcal{F} \subseteq \mathcal{V}$ is a Hilbert space that can be represented by using Φ with parameters from Θ. Function $\Pi : \mathcal{V} \to \mathcal{F}$ is a projection mapping, it renders a representation from $\mathcal{F}$ to each value function from $\mathcal{V}$. Operator $B_t : \mathcal{F} \to \mathcal{V}$ acts on (approximated) value functions. Finally, γ_t denotes the stepsize and W_t is a noise parameter representing the uncertainties coming from, for example, the simulation.

Operator B_t is time-dependent since, for example, if we model an approximate version of optimistic policy iteration, then in each iteration the control policy changes and, therefore, the update operator changes, as well. We can notice that if Π was a linear operator (see below), Equation (11) would be a stochastic iterative algorithm with $K_t = \Pi B_t$. Consequently, the algorithm described by Equation (6) is a generalization of many ADP methods, as well.

Now, we show that a convergence theorem for ADP methods can also be deduced by using Theorem 20. In order to apply the theorem, we should ensure that each update operator be a contraction. If we assume that every B_t is a contraction, we should require two properties from Π to guarantee that the resulted operators remain contractions. First, Π should be *linear*. Operator Π is linear if it is *additive* and *homogeneous*, more precisely, if $\forall V_1, V_2 : \Pi(V_1 + V_2) = \Pi(V_1) + \Pi(V_2)$ and $\forall V : \forall \alpha : \Pi(\alpha V) = \alpha \Pi(V)$, where α is a scalar. This requirement allows the separation of the components. Moreover, Π should be *nonexpansive* w.r.t. the supremum norm, namely: $\forall V_1, V_2 : \|\Pi(V_1) - \Pi(V_2)\| \leq \|V_1 - V_2\|$. Then, the update operator of the algorithm, $K_t = \Pi B_t$, is guaranteed to be a contraction.

If we assume that V_t^* is the fixed point of K_t, thus, $(\Pi B_t)V_t^* = V_t^*$ and β_t is the Lipschitz constant of K_t with $\limsup_{t\to\infty} \beta_t = \beta_0 < 1$, we can deduce a convergence theorem for ADP methods, as a corollary of Theorem 20. Suppose that Assumptions 1-2 hold and each B_t is a contraction as well as Π is linear and supremum norm nonexpansive, then $\Phi(r_t)$ κ-approximates V^* for any initial r_0 with $\kappa = 4\rho/(1-\beta_0)$, where $\rho = \limsup_{t\to\infty} \|V_t^* - V^*\|$. In case all of the fixed points were the same, viz., $\forall t : V_0^* = V_t^*$, then $\Phi(r_t)$ would converge to V_0^* almost surely, consequently, $\Phi(r_t)$ would κ-approximate V^* with $\kappa = \|V_0^* - V^*\|$.

Naturally, these results are quite loose, since we did not make strong assumptions on the applied algorithm and on the approximation architecture. They only illustrate that the approach we took, which allows time-dependent update operators and analyzes approximate convergence, could also provide results for ordinary MDPs, for example, in the case of ADP.

6. Experimental Results

In this section we present two numerical experiments. The first one demonstrates the effects of environmental changes during Q-learning based *scheduling*. The second one presents a parameter analysis concerning the effectiveness of SARSA in (ε, δ)-type *grid world* domains.

6.1 Environmental Changes During Scheduling

Scheduling is the allocation of *resources* over time to perform a collection of *jobs*. Each job consists of a set of *tasks*, potentially with precedence constraints, to be executed on the resources. The

job-shop scheduling problem (JSP) is one of the basic scheduling problems (Pinedo, 2002). We investigated an extension of JSP, called the *flexible job-shop* scheduling problem (FJSP), in which some of the resources are interchangeable, that is, there may be tasks that can be executed on several resources. This problem can be formulated as a finite horizon MDP and can be solved by Q-learning based methods (Csáji and Monostori, 2006).

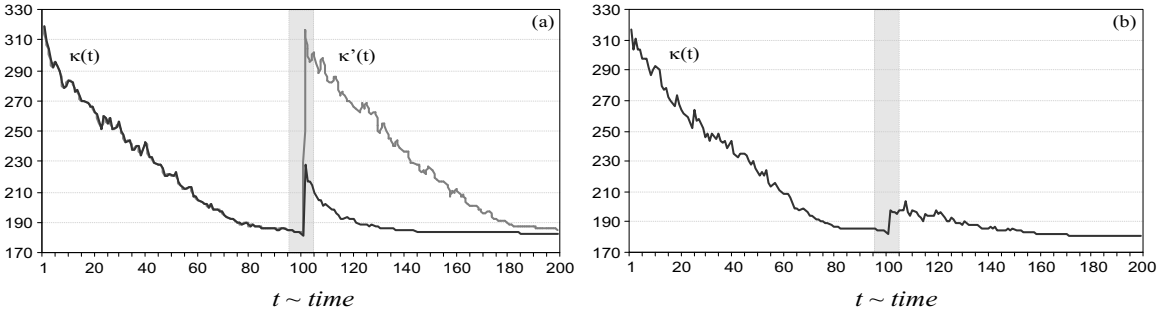

Figure 3: The black curves, $\kappa(t)$, show the performance measure in case there was a resource breakdown (a) or a new resource availability (b) at time $t = 100$; the gray curve in (a), $\kappa'(t)$, demonstrates the case the policy would be recomputed from scratch.

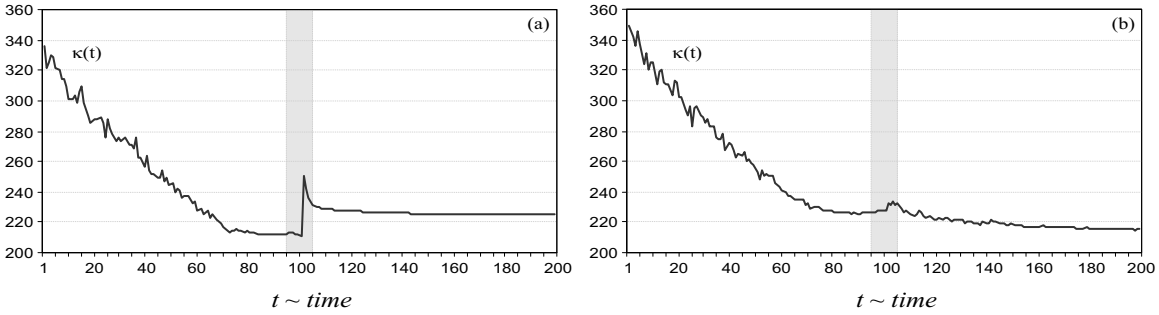

Figure 4: The black curves, $\kappa(t)$, show the performance measure during resource control in case there was a new job arrival (a) or a job cancellation (b) at time $t = 100$.

In order to investigate the effects of environmental changes during scheduling, numerical experiments were initiated and carried out. The aim of scheduling was to minimize the maximum completion time of the tasks, which performance measure is called "makespan". The adaptive features of the Q-learning based approach were tested by confronting the system with unexpected events, such as: resource breakdown, new resource availability (Figure 3), new job arrival or job cancellation (Figure 4). In Figures 3 and 4 the horizontal axis represents time, while the vertical one, the achieved performance measure. The figures were made by averaging hundred random samples. In these tests a fixed number of 20 resources were used with few dozens of jobs, where each job contained a sequence of tasks. In each case there was an unexpected event at time $t = 100$. After the change took place, we considered two possibilities: we either restarted the iterative scheduling process from scratch or we continued the learning using the current (obsolete) value function. We experienced that the latter approach is much more efficient. That was one of the reasons why we started to study how the optimal value function of an MDP depends on the dynamics of the system.

Recall that Theorems 10, 11 and 12 measure the amount of the possible change in the value function in case there were changes in the MDP, but since these theorems apply supremum norm, they only provide bounds for *worst case* situations. However, the results of our numerical experiments, shown in Figures 3 and 4, are indicative of the phenomenon that in an *average case* the change is much less. Therefore, applying the obsolete value function after a change took place is preferable over restarting the optimization from scratch.

The results, black curves, show the case when the obsolete value function approximation was applied after the change took place. The performance which would arise if the system recomputed the whole schedule from scratch is drawn in gray in part (a) of Figure 3.

6.2 Varying Grid World

We also performed numerical experiments on a variant of the classical *grid world* problem (Sutton and Barto, 1998). The original version of this problem can be briefly described as follows: an agent wanders in a rectangular world starting from a random *initial* state with the aim of finding the *goal* state. In each state the agent is allowed to choose from four possible actions: "north", "south", "east" and "west". After an action was selected, the agent moves one step in that direction. There are some *mines* on the field, as well, that the agent should avoid. An *episode* ends if the agent finds the goal state or hits a mine. During our experiments, we applied randomly generated 10×10 grid worlds (thus, these MDPs had 100 states) with 10 mines. The *immediate-cost* of taking a (non-terminating) step was 5, a cost of hitting a mine was 100 and the cost of finding the goal state was -100.

In order to perform the experiment described by Table 1, we have applied the "RL-Glue" framework[2] which consists of open source softwares and aims at being a standard protocol for benchmarking and interconnecting reinforcement learning agents and environments.

We have analyzed an (ε, δ)-type version of grid world, where the problem formed an (ε, δ)-MDP. More precisely, we have investigated the case when for all time t, the transition-probabilities could vary by at most $\varepsilon \geq 0$ around the base transition-probability values and the immediate-costs could vary by at most $\delta \geq 0$ around the base cost values.

During our numerical experiments, the environment changed at each time-step. These changes were generated as follows. First, changes concerning the transition-probabilities are described. In our randomized grid worlds the agent was taken to a random surrounding state (no matter what action it chose) with probability η and this probability *changed* after each step. The new η was computed according to the *uniform* distribution, but its possible values were *bounded* by the values described in the first row of Table 1.

Similarly, the immediate-costs of the base MDP (cf. the first paragraph) were *perturbed* with a uniform random variable that changed at each time-step. Again, its (absolute) value was bounded by δ, which is presented in the first column of the table. The values shown were divided by 100 to achieve the same scale as the transition-probabilities have.

Table 1 was generated using an (optimistic) SARSA algorithm, namely, the current policy was evaluated by SARSA, then the policy was (optimistically) improved, more precisely, the *greedy* policy with respect to the achieved evaluation was calculated. That policy was also *soft*, namely, it made random *explorations* with probability 0.05. We have generated 1000 random grid worlds for each parameter pairs and performed 10 000 episodes in each of these generated worlds. The results

2. RL-Glue can be found at http://rlai.cs.ualberta.ca/RLBB/top.html.

$\Delta\|g\|$	the bounds for the varying probability of arriving at random states $\sim \varepsilon$										
$\delta/100$	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
0.0	-55.5	-48.8	-41.4	-36.7	-26.7	-16.7	-8.5	2.1	14.2	31.7	46.0
0.1	-54.1	-46.1	-41.2	-34.5	-25.8	-15.8	-6.0	3.7	16.5	32.3	46.3
0.2	-52.5	-44.8	-40.1	-34.4	-25.3	-15.4	-5.8	4.0	17.6	33.1	48.1
0.3	-49.7	-42.1	-36.3	-31.3	-23.9	-14.2	-5.3	8.0	18.1	37.2	51.6
0.4	-47.4	-41.5	-34.7	-30.7	-22.2	-12.2	-2.3	8.8	20.2	38.3	52.0
0.5	-42.7	-41.0	-34.5	-24.8	-21.1	-10.1	-1.3	11.2	25.7	39.2	52.1
0.6	-36.1	-36.5	-29.7	-24.0	-16.8	-7.9	1.1	17.0	31.3	43.9	54.1
0.7	-30.2	-29.3	-29.3	-19.1	-13.4	-6.0	7.4	18.9	26.9	47.2	60.9
0.8	-23.1	-27.0	-21.4	-18.8	-10.9	-2.6	8.9	22.5	31.3	50.0	64.2
0.9	-14.1	-19.5	-21.0	-12.4	-7.5	0.7	13.2	23.2	38.9	52.2	68.1
1.0	-6.8	-10.7	-14.5	-7.1	-5.3	6.6	15.7	26.4	39.8	57.3	68.7

Table 1: The (average) cumulative costs gathered by SARSA in varying grid worlds.

presented in the table were calculated by averaging the cumulative costs over all episodes and over all generated sample worlds.

The parameter analysis shown in Table 1 is indicative of the phenomenon that changes in the transition-probabilities have a much higher impact on the performance. Even large perturbations in the costs were tolerated by SARSA, but large variations in the transition-probabilities caused a high decrease in the performance. An explanation could be that large changes in the transitions cause the agent to loose control over the events, since it becomes very hard to predict the effects of the actions and, hence, to estimate the expected costs.

7. Conclusion

The theory of MDPs provide a general framework for modeling decision making in stochastic dynamic systems, if we know a function that describes the dynamics or we can simulate it, for example, with a suitable program. In some situations, however, the dynamics of the system may change, too. In theory, this change can be modeled with another (higher level) MDP, as well, but doing so would lead to models which are practically intractable.

In the paper we have argued that the optimal value function of a (discounted) MDP Lipschitz continuously depends on the transition-probability function and the immediate-cost function, therefore, small changes in the environment result only in small changes in the optimal value function. This result was already known for the case of transition-probabilities, but we have presented an improved estimation for this case, as well. A bound for changes in the discount factor was also proven, and it was demonstrated that, in general, this dependence was not Lipschitz continuous. Additionally, it was shown that changes in the discount rate could be traced back to changes in the immediate-cost function. The application of the Lipschitz property helps the theoretical treatment of changing environments or inaccurate models, for example, if the transition-probabilities or the costs are estimated statistically, only.

In order to theoretically analyze environmental changes, the framework of (ε,δ)-MDPs was introduced as a generalization of classical MDPs and ε-MDPs. In this quasi-stationary model the

transition-probability function and the immediate-cost function may change over time, but the cumulative changes must remain bounded by ε and δ, asymptotically.

Afterwards, we have investigated how RL methods could work in this kind of changing environment. We have presented a general theorem that estimated the asymptotic distance of a value function sequence from a fixed value function. This result was applied to deduce a convergence theorem for value function based algorithms that work in (ε, δ)-MDPs.

In order to demonstrate our approach, we have presented some numerical experiments, too. First, two simple iterative processes were shown, a "well-behaving" stochastic process and a "pathological", oscillating deterministic process. Later, the effects of environmental changes on Q-learning based flexible job-shop scheduling was experimentally studied. Finally, we have analyzed how SARSA could work in varying (ε, δ)-type grid world domains.

We can conclude that value function based RL algorithms can work in varying environments, at least if the changes remain bounded in the limit. The asymptotic distance of the generated value function sequence from the optimal value function of the base MDP is bounded for a large class of stochastic iterative algorithms. Moreover, this bound is proportional to the diameter of this set, for example, to parameters ε and δ in the case of (ε, δ)-MDPs. These results were illustrated through three classical RL methods: asynchronous value iteration, Q-learning and temporal difference learning policy evaluation. We showed, as well, that this approach could be applied to investigate the convergence of ADP methods.

There are many potential further research directions. Now, as a conclusion to the paper, we highlight some of them. First, analyzing the effects of environmental changes on the value function in case of the *expected average cost* optimization criterion would be interesting. A promising direction could be to investigate environments with non-bounded changes, for example, when the environment might *drift* over time. Naturally, this drift should also be sufficiently slow in order to give the opportunity to the learning algorithm to *track* the changes. Another possible direction could be the further analysis of the convergence results in case of applying *value function approximation*. The classical problem of *exploration* and *exploitation* should also be reinvestigated in changing environments. Finally, for practical reasons, it would be important to find *finite time bounds* for the convergence of stochastic iterative algorithms for (a potentially restricted class of) non-stationary environments.

Acknowledgments

The work was supported by the Hungarian Scientific Research Fund (OTKA), Grant No. T73376, and by the EU-project Coll-Plexity, 12781 (NEST). Balázs Csanád Csáji greatly acknowledges the scholarship of the Hungarian Academy of Sciences. The authors are also very grateful to Csaba Szepesvári for the helpful comments and discussions.

Appendix A. Proofs

In this appendix the proofs of Theorems 11, 12, 13, 20 and Lemmas 14, 18 can be found.

Theorem 11 *Assume that two MDPs differ only in their transition-probability functions, denoted by p_1 and p_2. Let the corresponding optimal value functions be J_1^* and J_2^*, then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{\alpha \|g\|_\infty}{(1-\alpha)^2} \|p_1 - p_2\|_1 ,$$

where $\|\cdot\|_1$ is a norm on $f : \mathbb{X} \times \mathbb{A} \times \mathbb{X} \to \mathbb{R}$ type functions, for example, $f(x,a,y) = p(y\,|\,x,a)$,

$$\|f\|_1 = \max_{x,a} \sum_{y \in \mathbb{X}} |\, f(x,a,y)\,| .$$

Proof First, let us introduce a deterministic Markovian policy. For all state $x \in \mathbb{X}$:

$$\hat{\pi}(x) = \begin{cases} \arg\min\limits_{a \in \mathcal{A}(x)} \left[g(x,a) + \alpha \sum\limits_{y \in \mathbb{X}} p_1(y \mid x,a)\, J_1^*(y) \right] & \text{if } J_1^*(x) \leq J_2^*(x), \\ \arg\min\limits_{a \in \mathcal{A}(x)} \left[g(x,a) + \alpha \sum\limits_{y \in \mathbb{X}} p_2(y \mid x,a)\, J_2^*(y) \right] & \text{if } J_2^*(x) < J_1^*(x) \end{cases}$$

If the arg min is ambiguous then any action that takes the minimum can be selected. Using the Bellman optimality equation in the first step, $\|J_1^* - J_2^*\|_\infty$ can be estimated as follows,

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| =$$

$$= \left| \min_{a \in \mathcal{A}(x)} \left[g(x,a) + \alpha \sum_{y \in \mathbb{X}} p_1(y \mid x,a)\, J_1^*(y) \right] - \min_{a \in \mathcal{A}(x)} \left[g(x,a) + \alpha \sum_{y \in \mathbb{X}} p_2(y \mid x,a)\, J_2^*(y) \right] \right| \leq$$

$$\leq \left| g(x,\hat{\pi}(x)) + \alpha \sum_{y \in \mathbb{X}} p_1(y \mid x,\hat{\pi}(x))\, J_1^*(y) - g(x,\hat{\pi}(x)) - \alpha \sum_{y \in \mathbb{X}} p_2(y \mid x,\hat{\pi}(x))\, J_2^*(y) \right| ,$$

where we applied that $\forall\, f_1, f_2 : \mathcal{S} \to \mathbb{R}$ bounded functions such that $\min_s f_1(s) \leq \min_s f_2(s)$ and $\hat{s} = \arg\min_s f_1(s)$, we have $|\min_s f_1(s) - \min_s f_2(s)| \leq |\, f_1(\hat{s}) - f_2(\hat{s})|$. Then,

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| \leq \left| \alpha \sum_{y \in \mathbb{X}} p_1(y \mid x,\hat{\pi}(x)) J_1^*(y) - p_2(y \mid x,\hat{\pi}(x)) J_2^*(y) \right| =$$

$$= \left| \alpha \sum_{y \in \mathbb{X}} (p_1(y \mid x,\hat{\pi}(x)) - p_2(y \mid x,\hat{\pi}(x))) J_1^*(y) + \alpha \sum_{y \in \mathbb{X}} p_2(y \mid x,\hat{\pi}(x))(J_1^*(y) - J_2^*(y)) \right| \leq$$

$$\leq \alpha \sum_{y \in \mathbb{X}} |(p_1(y \mid x,\hat{\pi}(x)) - p_2(y \mid x,\hat{\pi}(x))) J_1^*(y)| + \alpha \sum_{y \in \mathbb{X}} |p_2(y \mid x,\hat{\pi}(x))(J_1^*(y) - J_2^*(y))|,$$

where in the second step we have rewritten $p_1(y|x,\hat{\pi}(x))J_1^*(y) - p_2(y|x,\hat{\pi}(x))J_2^*(y)$ as

$$p_1(y|x,\hat{\pi}(x))J_1^*(y) - p_2(y|x,\hat{\pi}(x))J_2^*(y) =$$

$$= p_1(y|x,\hat{\pi}(x))J_1^*(y) - p_2(y|x,\hat{\pi}(x))J_1^*(y) + p_2(y|x,\hat{\pi}(x))J_1^*(y) - p_2(y|x,\hat{\pi}(x))J_2^*(y) =$$

$$= (p_1(y\,|\,x,\hat{\pi}(x)) - p_2(y\,|\,x,\hat{\pi}(x)))J_1^*(y) + p_2(y\,|\,x,\hat{\pi}(x))(J_1^*(y) - J_2^*(y)).$$

Now, let us recall (a special form of) *Hölder's inequality*: let v_1, v_2 be two vectors and $1 \leq q, r \leq \infty$ with $1/q + 1/r = 1$. Then, we have $\|v_1 v_2\|_{(1)} \leq \|v_1\|_{(q)} \|v_2\|_{(r)}$, where $\|\cdot\|_{(q)}$ denotes *vector* norm, for example, $\|v\|_{(q)} = (\sum_i |v_i|^q)^{1/q}$ and $\|v\|_{(\infty)} = \max_i |v_i| = \|v\|_\infty$. Here, we applied the unusual "(q)" notation to avoid confusion with the applied matrix norm. Notice that the first sum of the last estimation can be treated as the (1)-norm of $v_1 v_2$, where

$$v_1(y) = p_1(y \mid x,\hat{\pi}(x)) - p_2(y \mid x,\hat{\pi}(x))) \quad \text{and} \quad v_2(y) = J_1^*(y),$$

after which Hölder's inequality can be applied with $q = 1$ and $r = \infty$ to estimate the sum. A similar argument can be repeated in the case of the second sum with

$$v_1(y) = p_2(y \mid x,\hat{\pi}(x)) \quad \text{and} \quad v_2(y) = J_1^*(y) - J_2^*(y).$$

Then, after the two applications of Hölder's inequality, we have for all x that

$$|J_1^*(x) - J_2^*(x)| \leq \alpha \|p_1(\cdot \mid x,\hat{\pi}(x)) - p_2(\cdot \mid x,\hat{\pi}(x))\|_{(1)} \|J_1^*\|_\infty +$$

$$+ \alpha \|p_2(\cdot \mid x,\hat{\pi}(x))\|_{(1)} \|J_1^* - J_2^*\|_\infty,$$

since $\|J_1^*\|_\infty \leq \|g\|_\infty/(1-\alpha)$, $\|p_2(\cdot \mid x,\hat{\pi}(x))\|_{(1)} = 1$ and we have this estimation for all x,

$$\|J_1^* - J_2^*\|_\infty \leq \frac{\alpha\|g\|_\infty}{1-\alpha} \max_{x\in\mathbb{X}} \sum_{y\in\mathbb{X}} |\,p_1(y\,|\,x,\hat{\pi}(x)) - p_2(y\,|\,x,\hat{\pi}(x))\,| + \alpha\|J_1^* - J_2^*\|_\infty,$$

which formula can be overestimated, by taking the maximum over all actions, by

$$\|J_1^* - J_2^*\|_\infty \leq \frac{\alpha\|g\|_\infty}{1-\alpha} \|p_1 - p_2\|_1 + \alpha\|J_1^* - J_2^*\|_\infty,$$

from which the statement of the theorem immediately follows after rearrangement. ■

Theorem 12 *Assume that two discounted MDPs differ only in the immediate-cost functions, denoted by g_1 and g_2. Let the corresponding optimal value functions be J_1^* and J_2^*, then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{1}{1-\alpha} \|g_1 - g_2\|_\infty.$$

Proof First, let us introduce a deterministic Markovian policy. For all state $x \in \mathbb{X}$:

$$\hat{\pi}(x) = \begin{cases} \displaystyle\operatorname*{arg\,min}_{a\in\mathcal{A}(x)} \left[g_1(x,a) + \alpha \sum_{y\in\mathbb{X}} p(y \mid x,a) J_1^*(y) \right] & \text{if } J_1^*(x) \leq J_2^*(x), \\ \displaystyle\operatorname*{arg\,min}_{a\in\mathcal{A}(x)} \left[g_2(x,a) + \alpha \sum_{y\in\mathbb{X}} p(y \mid x,a) J_2^*(y) \right] & \text{if } J_2^*(x) < J_1^*(x). \end{cases}$$

If the arg min is ambiguous, then any action that takes the minimum can be selected. Using the Bellman optimality equation in the first step, $\|J_1^* - J_2^*\|_\infty$ can be estimated as follows,

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| =$$

$$= \left| \min_{a \in \mathcal{A}(x)} \left[g_1(x,a) + \alpha \sum_{y \in \mathbb{X}} p(y \mid x,a) J_1^*(y) \right] - \min_{a \in \mathcal{A}(x)} \left[g_2(x,a) + \alpha \sum_{y \in \mathbb{X}} p(y \mid x,a) J_2^*(y) \right] \right| \leq$$

$$\leq \left| g_1(x,\hat{\pi}(x)) + \alpha \sum_{y \in \mathbb{X}} p(y \mid x,\hat{\pi}(x)) J_1^*(y) - g_2(x,\hat{\pi}(x)) - \alpha \sum_{y \in \mathbb{X}} p(y \mid x,\hat{\pi}(x)) J_2^*(y) \right|,$$

where we applied that $\forall f_1, f_2 : \mathcal{S} \to \mathbb{R}$ bounded functions such that $\min_s f_1(s) \leq \min_s f_2(s)$ and $\hat{s} = \arg\min_s f_1(s)$, we have $|\min_s f_1(s) - \min_s f_2(s)| \leq |f_1(\hat{s}) - f_2(\hat{s})|$. Then,

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| \leq |g_1(x,\hat{\pi}(x)) - g_2(x,\hat{\pi}(x))| + \alpha \sum_{y \in \mathbb{X}} p(y \mid x,\hat{\pi}(x)) \, |J_1^*(y) - J_2^*(y)| \leq$$

$$\leq \|g_1 - g_2\|_\infty + \alpha \sum_{y \in \mathbb{X}} p(y \mid x,\hat{\pi}(x)) \, \|J_1^* - J_2^*\|_\infty =$$

$$= \|g_1 - g_2\|_\infty + \alpha \|J_1^* - J_2^*\|_\infty .$$

It is easy to see that if

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| \leq \|g_1 - g_2\|_\infty + \alpha \|J_1^* - J_2^*\|_\infty ,$$

then

$$\|J_1^* - J_2^*\|_\infty \leq \|g_1 - g_2\|_\infty + \alpha \|J_1^* - J_2^*\|_\infty ,$$

from which the statement of the theorem immediately follows after rearrangement. ■

Theorem 13 *Assume that two discounted MDPs differ only in the discount factors, denoted by* $\alpha_1, \alpha_2 \in [0,1)$. *Let the corresponding optimal value functions be* J_1^* *and* J_2^*, *then*

$$\|J_1^* - J_2^*\|_\infty \leq \frac{|\alpha_1 - \alpha_2|}{(1-\alpha_1)(1-\alpha_2)} \|g\|_\infty .$$

Proof Let π_i^* denote a greedy and deterministic policy based on value function J_i^*, where $i \in \{1,2\}$. Naturally, policy π_i^* is optimal if the discount rate is α_i (Theorem 6). Then, let us introduce a deterministic Markovian control policy $\hat{\pi}$ defined as

$$\hat{\pi}(x) = \begin{cases} \pi_1^*(x) & \text{if } J_1^*(x) \leq J_2^*(x), \\ \\ \pi_2^*(x) & \text{if } J_2^*(x) < J_1^*(x). \end{cases}$$

For any state x the difference of the two value functions can be estimated as follows,

$$|J_1^*(x) - J_2^*(x)| =$$

$$= \left| \min_{a\in\mathcal{A}(x)} \left[g(x,a) + \alpha_1 \sum_{y\in\mathbb{X}} p(y \mid x,a)\, J_1^*(y) \right] - \min_{a\in\mathcal{A}(x)} \left[g(x,a) + \alpha_2 \sum_{y\in\mathbb{X}} p(y \mid x,a)\, J_2^*(y) \right] \right| \leq$$

$$\leq \left| g(x,\hat{\pi}(x)) + \alpha_1 \sum_{y\in\mathbb{X}} p(y \mid x,\hat{\pi}(x))\, J_1^*(y) - g(x,\hat{\pi}(x)) - \alpha_2 \sum_{y\in\mathbb{X}} p(y \mid x,\hat{\pi}(x))\, J_2^*(y) \right|,$$

where we applied that $\forall\, f_1, f_2 : \mathcal{S} \to \mathbb{R}$ bounded functions such that $\min_s f_1(s) \leq \min_s f_2(s)$ and $\hat{s} = \arg\min_s f_1(s)$, we have $|\min_s f_1(s) - \min_s f_2(s)| \leq |f_1(\hat{s}) - f_2(\hat{s})|$. Then,

$$\forall x \in \mathbb{X} : |J_1^*(x) - J_2^*(x)| \leq \left| \sum_{y\in\mathbb{X}} p(y \mid x,\hat{\pi}(x)) \left(\alpha_1 J_1^*(y) - \alpha_2 J_2^*(y)\right) \right| \leq$$

$$\leq |\alpha_1 - \alpha_2| \frac{1}{1-\alpha_1} \|g\|_\infty + \alpha_2 \|J_1^* - J_2^*\|_\infty,$$

where in the last step we used the following estimation of $|\alpha_1 J_1^*(y) - \alpha_2 J_2^*(y)|$,

$$|\alpha_1 J_1^*(y) - \alpha_2 J_2^*(y)| = |\alpha_1 J_1^*(y) - \alpha_2 J_1^*(y) + \alpha_2 J_1^*(y) - \alpha_2 J_2^*(y)| \leq$$

$$\leq |\alpha_1 - \alpha_2|\, |J_1^*(y)| + \alpha_2\, |J_1^*(y) - J_2^*(y)| \leq |\alpha_1 - \alpha_2| \frac{1}{1-\alpha_1} \|g\|_\infty + \alpha_2 \|J_1^* - J_2^*\|_\infty,$$

where we applied the fact that for any state y we have,

$$|J_1^*(y)| \leq \sum_{t=0}^{\infty} \alpha_1^t \|g\|_\infty = \frac{1}{1-\alpha_1} \|g\|_\infty.$$

Because the estimation of $|J_1^*(x) - J_2^*(x)|$ is valid for all x, we have the following result

$$\|J_1^* - J_2^*\|_\infty \leq |\alpha_1 - \alpha_2| \frac{1}{1-\alpha_1} \|g\|_\infty + \alpha_2 \|J_1 - J_2\|_\infty,$$

from which the statement of the theorem immediately follows after rearrangement. ■

Lemma 14 *Assume that we have two discounted MDPs which differ only in the transition-probability functions or only in the immediate-cost functions or only in the discount factors. Let the corresponding optimal action-value functions be Q_1^* and Q_2^*, respectively. Then, the bounds for $\|J_1^* - J_2^*\|_\infty$ of Theorems 11, 12 and 13 are also bounds for $\|Q_1^* - Q_2^*\|_\infty$.*

Proof We will prove the theorem in three parts, depending on the changing components.
Case 1: Assume that the MDPs differ only in the transition functions, denoted by p_1 and p_2. We will prove the same estimation as in the case of Theorem 11, more precisely, that

$$\|Q_1^* - Q_2^*\|_\infty \leq \frac{\alpha \|g\|_\infty}{(1-\alpha)^2} \|p_1 - p_2\|_1.$$

For all state-action pair (x,a) we can estimate the absolute difference of Q_1^* and Q_2^* as

$$|Q_1^*(x,a) - Q_2^*(x,a)| =$$

$$= \left| g(x,a) + \alpha \sum_{y\in\mathbb{X}} p_1(y|x,a)J_1^*(y) - g(x,a) - \alpha \sum_{y\in\mathbb{X}} p_2(y|x,a)J_2^*(y) \right| \leq$$

$$\leq \left| \alpha \sum_{y\in\mathbb{X}} \left(p_1(y|x,a)J_1^*(y) - p_2(y|x,a)J_2^*(y)\right) \right|,$$

from which the proof continues in the same way as the proof of Theorem 11.
Case 2: Assume that the MDPs differ only in the immediate-cost functions, denoted by g_1 and g_2. We will prove the same estimation as in the case of Theorem 12, more precisely,

$$\|Q_1^* - Q_2^*\|_\infty \leq \frac{1}{1-\alpha} \|g_1 - g_2\|_\infty .$$

For all state-action pair (x,a) we can estimate the absolute difference of Q_1^* and Q_2^* as

$$|Q_1^*(x,a) - Q_2^*(x,a)| =$$

$$= \left| g_1(x,a) + \alpha \sum_{y\in\mathbb{X}} p(y|x,a)J_1^*(y) - g_2(x,a) - \alpha \sum_{y\in\mathbb{X}} p(y|x,a)J_2^*(y) \right| \leq$$

$$\leq \|g_1 - g_2\|_\infty + \left| \alpha \sum_{y\in\mathbb{X}} p(y|x,a)(J_1^*(y) - J_2^*(y)) \right| \leq \|g_1 - g_2\|_\infty + \alpha \|J_1^* - J_2^*\|_\infty .$$

The statement immediately follows after we apply Theorem 12 to estimate $\|J_1^* - J_2^*\|_\infty$.
Case 3: Assume that the MDPs differ only in the discount rates, denoted by α_1 and α_2. We will prove the same estimation as in the case of Theorem 13, more precisely, that

$$\|Q_1^* - Q_2^*\|_\infty \leq \frac{|\alpha_1 - \alpha_2|}{(1-\alpha_1)(1-\alpha_2)} \|g\|_\infty .$$

For all state-action pair (x,a) we can estimate the absolute difference of Q_1^* and Q_2^* as

$$|Q_1^*(x,a) - Q_2^*(x,a)| =$$

$$= \left| g(x,a) + \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a)J_1^*(y) - g(x,a) - \alpha_2 \sum_{y\in\mathbb{X}} p(y|x,a)J_2^*(y) \right| \leq$$

$$\leq \left| \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a)J_1^*(y) - \alpha_2 \sum_{y\in\mathbb{X}} p(y|x,a)J_2^*(y) \right| \leq |\alpha_1 - \alpha_2| \frac{1}{1-\alpha_1} \|g\|_\infty + \alpha_2 \|J_1^* - J_2^*\|_\infty ,$$

where in the last step we applied the same estimation as in the proof of Theorem 13. The statement immediately follows after we apply Theorem 13 to estimate $\|J_1^* - J_2^*\|_\infty$. ■

Lemma 18 *Assume that two discounted MDPs, $\mathcal{M}_1$ and $\mathcal{M}_2$, differ only in the discount factors, denoted by α_1 and α_2. Then, there exists an MDP, denoted by $\mathcal{M}_3$, such that it differs only in the immediate-cost function from $\mathcal{M}_1$, thus its discount factor is α_1, and it has the same optimal value function as $\mathcal{M}_2$. The immediate-cost function of $\mathcal{M}_3$ is*

$$\widehat{g}(x,a) = g(x,a) + (\alpha_2 - \alpha_1) \sum_{y\in\mathbb{X}} p(y|x,a)J_2^*(y),$$

where p is the probability-transition function of $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$; g is the immediate-cost function of $\mathcal{M}_1$ and $\mathcal{M}_2$; and $J_2^*(y)$ denotes the optimal cost-to-go function of $\mathcal{M}_2$.

Proof First of all, let us overview some general statements that will be used in the proof.

Recall from Bertsekas and Tsitsiklis (1996) that we can treat the solution (the optimal value function) of the infinite horizon problem as the limit of the finite horizon solutions. More precisely, the Bellman optimality equation for the n-stage (finite horizon) problem is

$$J_k^*(x) = \min_{a\in\mathcal{A}(x)} \Big[g(x,a) + \alpha \sum_{y\in\mathbb{X}} p(y \mid x,a)\, J_{k-1}^*(y) \Big],$$

for all $k \in \{1,\ldots,n\}$ and $x \in \mathbb{X}$. Note that by definition, we have $J_0^*(x) = 0$. Moreover,

$$\forall x \in \mathbb{X} : J^*(x) = J_\infty^*(x) = \lim_{n\to\infty} J_n^*(x).$$

Also recall that the n-stage optimal action value function is defined as

$$Q_k^*(x,a) = g(x,a) + \alpha \sum_{y\in\mathbb{X}} p(y|x,a) J_{k-1}^*(y),$$

for all x, a and $k \in \{1,\ldots,n\}$. We also have $Q_0^*(x,a) = 0$ and $J_n^*(x) = \min_a Q_n^*(x,a)$.

During the proof we will apply the solutions of suitable finite horizon problems, thus, in order to avoid notational confusions, let us denote the optimal state and action value functions of $\mathcal{M}_2$ and $\mathcal{M}_3$ by J^*, Q^* and $\hat{J}^*$, $\widehat{Q}^*$, respectively. The corresponding finite horizon optimal value functions will be denoted by J_n^*, Q_n^* and $\hat{J}_n^*$, $\widehat{Q}_n^*$, respectively, where n is the length of the horizon. We will show that for all state x and action a we have $Q^*(x,a) = \widehat{Q}^*(x,a)$, from which $J^* = \hat{J}^*$ follows. Let us define $\widehat{g}_n$ for all $n > 0$ by

$$\widehat{g}_n(x,a) = g(x,a) + (\alpha_2 - \alpha_1) \sum_{y\in\mathbb{X}} p(y|x,a) J_{n-1}^*(y).$$

We will apply induction on n. For the case of $n = 0$ we trivially have $Q_0^* = \widehat{Q}_0^*$, since both of them are constant zero functions. Now, assume that $Q_k^* = \widehat{Q}_k^*$ for $k \le n$, then

$$\widehat{Q}_{n+1}^*(x,a) = \widehat{g}_{n+1}(x,a) + \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a) \hat{J}_n^*(y) =$$

$$= g(x,a) + (\alpha_2 - \alpha_1) \sum_{y\in\mathbb{X}} p(y|x,a) J_n^*(y) + \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a) \hat{J}_n^*(y) =$$

$$= g(x,a) + \alpha_2 \sum_{y\in\mathbb{X}} p(y|x,a) J_n^*(y) + \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a) \big(\hat{J}_n^*(y) - J_n^*(y) \big) =$$

$$= g(x,a) + \alpha_2 \sum_{y\in\mathbb{X}} p(y|x,a) J_n^*(y) + \alpha_1 \sum_{y\in\mathbb{X}} p(y|x,a) \left(\min_{b\in\mathcal{A}(y)} \widehat{Q}_n^*(y,b) - \min_{b\in\mathcal{A}(y)} Q_n^*(y,b) \right) =$$

$$= g(x,a) + \alpha_2 \sum_{y\in\mathbb{X}} p(y|x,a) J_n^*(y) = Q_{n+1}^*(x,a).$$

We have proved that for all n: $Q_n^* = \widehat{Q}_n^*$. Consequently, $Q^*(x,a) = \lim_{n\to\infty} Q_n^*(x,a) = \lim_{n\to\infty} \widehat{Q}_n^*(x,a)$ $= \widehat{Q}^*(x,a)$ and, thus, $J^*(x) = \min_a Q^*(x,a) = \min_a \widehat{Q}^*(x,a) = \hat{J}^*(x)$. Finally, note that for the case

of the infinite horizon problem $\widehat{g}(x,a) = \lim_{n\to\infty} \widehat{g}_n(x,a)$. ■

Theorem 20 *Suppose that Assumptions 1-3 hold and let V_t be the sequence generated by*

$$V_{t+1}(x) = (1-\gamma_t(x))V_t(x) + \gamma_t(x)((K_tV_t)(x) + W_t(x)),$$

then, for any $V^*, V_0 \in \mathcal{V}$, the sequence V_t κ-approximates function V^* with

$$\kappa = \frac{4\rho}{1-\beta_0} \quad \textit{where} \quad \rho = \limsup_{t\to\infty} \|V_t^* - V^*\|_\infty.$$

The applied three main assumptions are as follows
Assumption 1 *There exits a constant $C > 0$ such that for all state x and time t, we have*

$$\mathbb{E}[W_t(x) \,|\, \mathcal{F}_t] = 0 \quad \textit{and} \quad \mathbb{E}\left[W_t^2(x) \,|\, \mathcal{F}_t\right] < C < \infty.$$

Assumption 2 *For all x and t, $0 \le \gamma_t(x) \le 1$, and we have with probability one*

$$\sum_{t=0}^{\infty} \gamma_t(x) = \infty \quad \textit{and} \quad \sum_{t=0}^{\infty} \gamma_t^2(x) < \infty.$$

Assumption 3 *For all t, operator $K_t : \mathcal{V} \to \mathcal{V}$ is a supremum norm contraction mapping with Lipschitz constant $\beta_t < 1$ and with fixed point V_t^*. Formally, for all $V_1, V_2 \in \mathcal{V}$,*

$$\|K_tV_1 - K_tV_2\|_\infty \le \beta_t \|V_1 - V_2\|_\infty.$$

Let us introduce a common Lipschitz constant $\beta_0 = \limsup_{t\to\infty} \beta_t$, and assume that $\beta_0 < 1$.

Proof During the proof, our main aim will be to apply Theorem 9, thus, we have to show that the assumptions of the theorem hold. Let us define operator H_t for all $V_a, V_b \in \mathcal{V}$ by

$$H_t(V_a, V_b)(x) = (1-\gamma_t(x))V_a(x) + \gamma_t(x)((K_tV_b)(x) + W_t(x)).$$

Applying this definition, first, we will show that $V'_{t+1} = H_t(V'_t, V^*)$ κ-approximates V^* for all V'_0. Because $\beta_t < 1$ for all t and $\limsup_{t\to\infty} \beta_t = \beta_0 < 1$, it follows that $\sup_t \beta_t = \widetilde{\beta} < 1$ and each K_t is $\widetilde{\beta}$ contraction. We know that $\limsup_{t\to\infty} \|V^* - V_t^*\|_\infty = \rho$, therefore, for all $\delta > 0$, there is an index t_0 such that for all $t \ge t_0$, we have that $\|V^* - V_t^*\|_\infty \le \rho + \delta$. Using these observations, we can estimate $\|K_tV^*\|_\infty$ for all $t > t_0$, as follows

$$\|K_tV^*\|_\infty = \|K_tV^* - V^* + V^*\|_\infty \le \|K_tV^* - V^*\|_\infty + \|V^*\|_\infty \le$$

$$\le \|K_tV^* - V_t^* + V_t^* - V^*\|_\infty + \|V^*\|_\infty \le \|K_tV^* - V_t^*\|_\infty + \|V_t^* - V^*\|_\infty + \|V^*\|_\infty \le$$

$$\le \|K_tV^* - K_tV_t^*\|_\infty + \rho + \delta + \|V^*\|_\infty \le \widetilde{\beta}\|V^* - V_t^*\|_\infty + \rho + \delta + \|V^*\|_\infty \le$$

$$\le (1+\widetilde{\beta})\rho + (1+\widetilde{\beta})\delta + \|V^*\|_\infty \le (1+\widetilde{\beta})\rho + 2\delta + \|V^*\|_\infty.$$

If we apply $\delta = (1-\widetilde{\beta})\rho/2$, then for sufficiently large t $(t \ge t_0)$ we have that

$$\|K_tV^*\|_\infty \le 2\rho + \|V^*\|_\infty.$$

Now, we can upper estimate $V'_{t+1} = H_t(V'_t, V^*)$, for all $x \in \mathcal{X}$, $V'_0 \in \mathcal{V}$ and $t \geq t_0$ by

$$V'_{t+1}(x) = H_t(V'_t, V^*)(x) = (1-\gamma_t(x))V'_t(x) + \gamma_t(x)((K_t V^*)(x) + W_t(x)) \leq$$

$$\leq (1-\gamma_t(x))V'_t(x) + \gamma_t(x)(\|K_t V^*\|_\infty + W_t(x)) \leq$$

$$\leq (1-\gamma_t(x))V'_t(x) + \gamma_t(x)(\|V^*\|_\infty + 2\rho + W_t(x)).$$

Let us define a new sequence for all $x \in \mathcal{X}$ by

$$\widetilde{V}_{t+1}(x) = \begin{cases} (1-\gamma_t(x))\widetilde{V}_t(x) + \gamma_t(x)(\|V^*\|_\infty + 2\rho + W_t(x)) & \text{if } t \geq t_0, \\ V'_t(x) & \text{if } t < t_0. \end{cases}$$

It is easy to see (for example, by induction from t_0) that for all time t and state x we have that $V'_t(x) \leq \widetilde{V}_t(x)$ with probability one, more precisely, for almost all $\omega \in \Omega$, where $\omega = \langle \omega_1, \omega_2, \dots \rangle$ drives the noise parameter $W_t(x) = w_t(x, \omega_t)$ in both V'_t and $\widetilde{V}_t$. Note that Ω is the sample space of the underlying probability measure space $\langle \Omega, \mathcal{F}, \mathbb{P} \rangle$.

Applying the "Conditional Averaging Lemma" of Szepesvári and Littman (1999), which is a variant of the Robbins-Monro Theorem and requires Assumptions 1 and 2, we get that $\widetilde{V}_t(x)$ converges with probability one to $2\rho + \|V^*\|_\infty$ for all $\widetilde{V}_0 \in \mathcal{V}$ and $x \in \mathcal{X}$. Therefore, because $V'_t(x) \leq \widetilde{V}_t(x)$ for all x and t with probability one, we have that the sequence $V'_t(x)$ κ-approximates $V^*(x)$ with $\kappa = 2\rho$ for all function $V'_0 \in \mathcal{V}$ and state $x \in \mathcal{X}$.

Now, let us turn to Conditions 1-4 of Theorem 9. For all x and t, we define functions $F_t(x)$ and $G_t(x)$ as $F_t(x) = \beta_t \gamma_t(x)$ and $G_t(x) = (1-\gamma_t(x))$. By Assumption 2, functions $F_t(x), G_t(x) \in [0,1]$ for all x and t. Condition 1 trivially follows from the definitions of G_t and H_t. For the proof of Condition 2, we need Assumption 3, namely that each operator K_t is a contraction with respect to β_t. Condition 3 is a consequence of Assumption 2 and the definition of G_t. Finally, we have Condition 4 for any $\varepsilon > 0$ and sufficiently large t by defining $\xi = \beta_0 + \varepsilon$. Applying Theorem 9 with $\kappa = 2\rho$, we get that V_t κ'-approximates V^* with $\kappa' = 4\rho/(1-\beta_0-\varepsilon)$. In the end, letting ε go to zero proves our statement. ■

References

D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, Belmont, Massachusetts, 3rd edition, 2007.

D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

B. Cs. Csáji. *Adaptive Resource Control: Machine Learning Approaches to Resource Allocation in Uncertain and Changing Environments*. PhD thesis, Faculty of Informatics, Eötvös Loránd University, Budapest, 2008.

B. Cs. Csáji and L. Monostori. Adaptive sampling based large-scale stochastic resource control. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-06), July 16-20, Boston, Massachusetts*, pages 815–820, 2006.

R. Montes de Oca, A. Sakhanenko, and F. Salem. Estimates for perturbations of general discounted Markov control chains. *Applied Mathematics*, 30:287–304, 2003.

E. Even-Dar and Y. Mansour. Learning rates for Q-learning. *Journal of Machine Learning Research (JMLR)*, 5:1–25, Dec. 2003.

G. Favero and W. J. Runggaldier. A robustness result for stochastic control. *Systems and Control Letters*, 46:91–66, 2002.

E. A. Feinberg and A. Shwartz, editors. *Handbook of Markov Decision Processes: Methods and Applications*. Kluwer Academic Publishers, 2002.

E. Gordienko and F. S. Salem. Estimates of stability of Markov control processes with unbounded cost. *Kybernetika*, 36:195–210, 2000.

Zs. Kalmár, Cs. Szepesvári, and A. Lőrincz. Module-based reinforcement learning: Experiments with a real robot. *Machine Learning*, 31:55–85, 1998.

M. Kearns and S. Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49:209–232, 2002.

A. Müller. How does the solution of a Markov decision process depend on the transition probabilities? Technical report, Institute for Economic Theory and Operations Research, University of Karlsruhe, 1996.

R. Munos and A. W. Moore. Rates of convergence for variable resolution schemes in optimal control. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*, pages 647–654. Morgan Kaufmann, San Francisco, CA, 2000.

M. Pinedo. *Scheduling: Theory, Algorithms, and Systems*. Prentice-Hall, 2002.

S. Singh and D. Bertsekas. Reinforcement learning for dynamic channel allocation in cellular telephone systems. In *Advances in Neural Information Processing Systems*, volume 9, pages 974–980. The MIT Press, 1997.

R. S. Sutton and A. G. Barto. *Reinforcement Learning*. The MIT Press, 1998.

R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in Neural Information Processing Systems*, 12: 1057–1063, 2000.

Cs. Szepesvári and M. L. Littman. A unified analysis of value-function-based reinforcement learning algorithms. *Neural Computation*, 11(8):2017–2060, 1999.

I. Szita, B. Takács, and A. Lőrincz. ε-MDPs: Learning in varying environments. *Journal of Machine Learning Research (JMLR)*, 3:145–174, 2002.

B. Van Roy, D. Bertsekas, Y. Lee, and J. Tsitsiklis. A neuro-dynamic programming approach to retailer inventory management. Technical report, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA., 1996.