Proceedings of the 42nd IEEE
Conference on Decision and Control
Maui, Hawaii USA, December 2003

A System Theoretic Perspective of Learning and Optimization

Xi-Ren Cao*
Hong Kong University of Science and
Technology
Clear Water Bay, Kowloon, Hong Kong
eecao@ee.ust.hk

***Abstract*— Learning and optimization of stochastic systems is a multi-disciplinary area that attracts wide attentions from researchers in control systems, operations research and computer science. Areas such as perturbation analysis (PA), Markov decision process (MDP), and reinforcement learning (RL) share the common goal. In this paper, we offer an overview of the area of learning and optimization from a system theoretic perspective. We show how these seemly different disciplines are closely related, how one topic leads to the others, and how this perspective may lead to new research topics and new results, and how the performance sensitivity formulas can serve as the basis for learning and optimization.**

I. INTRODUCTION

Perturbation analysis [12] was originally developed for estimating performance derivatives with respect to system parameters in stochastic systems with queueing structures (queueing networks, generalized semi-Markov processes, etc); the estimates can be obtained by analyzing a single sample path of such a system; it was shown that although the approach requires some conditions for the system structure, it is very efficient since it utilizes the special dynamic properties of the system. The fundamental concept of PA, perturbation realization [5], has been extended to Markov processes. Recent research in this direction reveals a strong connection among PA, MDP(Markov decision processes), and RL (reinforcement learning) [6].

In this paper, we offer an overview of learning and optimization from a system theoretic perspective. We show how these seemly different disciplines are closely related, how one topic leads to the others, and how this perspective may lead to new research topics and new results. Our discussion is based on the general model of discrete time Markov chains. For simplicity, we discuss Markov chains with finite state space denoted as $\{1, 2, \cdots, M\}$. The central piece of learning and optimization is the performance potentials $g(i)$, $i = 1, \cdots, M$, or equivalently, perturbation realization factors $d(i, j) = g(j) - g(i)$ [7]. From perturbation analysis point of view, a change in system parameters induces a series of perturbations on a sample path, the effect of a single perturbation on a system performance can be measured by the realization factor of the perturbation, and the total effect of the parameter change on the performance is then the sum of the realization factors of all the perturbations induced by the parameter changes [5]. For Markov chains, parameters are the transition probabilities, a perturbation is a "jump" from one state i to another state j, and the realization factor equals the difference of the potentials at the two states. It has been shown that by the above principle, we can use potentials or realization factors as building blocks to construct performance sensitivities for many systems. When the changes are discrete, this leads to formulas for the performance difference of two Markov chains, and when the changes are infinitesimal, it leads to the formula for performance gradients [7].

These two standard formulas are the basis for performance optimization [6]: Optimization can be achieved by combining the gradient estimate with stochastic approximation methods, or by policy iteration which can be easily derived from the performance difference formula (see section II). This leads to the following main research directions:

1) Develop efficient algorithms to estimate the potentials and/or the derivatives. Reinforcement learning, TD(λ), neuro-dynamic programming, etc, are efficient ways of estimating the performance potentials, realization factors, and related quantities such as Q-factors, etc., based on sample paths (section III-A). In addition, algorithms can be developed to estimate performance gradients directly from a single sample path.
2) Develop efficient optimization algorithms with the potential or gradient estimates
   a) Gradient-based optimization for parameterized systems; this approach combines the gradient estimates with stochastic gradient algorithms.
   b) On-line policy iteration; this approach combines the potential estimates with stochastic approximation to implement policy iteration (section V).
   c) Gradient-based policy iteration; this is an open problem.

II. A GENERAL VIEW OF OPTIMIZATION

Consider an irreducible and aperiodic Markov chain $\mathbf{X} = \{X_n : n \geq 0\}$ on a finite state space $\mathcal{S} = \{1, 2, \cdots, M\}$ with transition probability matrix $P = [p(i, j)] \in [0, 1]^{M \times M}$. Let $\pi = (\pi_1, \ldots, \pi_M)$ be the (row) vector representing its steady-state probabilities, and $f = (f_1, f_2, \cdots, f_M)^T$ be the (column) performance vector, where "T" represents transpose. We have $Pe = e$, where $e = (1, 1, \cdots, 1)^T$ is an M-dimensional vector whose all components equal 1, and

*Supported in part by a grant from Hong Kong UGC. Tel: (852) 2358-7048 Fax: (852) 2358-1485

$\pi e = 1$. The steady state probability flow balance equation is $\pi = \pi P$. The performance measure is the long-run average defined as

$$\eta = E_\pi(f) = \sum_{i=1}^{M} \pi_i f_i = \pi f = \lim_{L\to\infty} \frac{1}{L} \sum_{l=0}^{L-1} f(X_l), w.p.1.$$

We start with the *Poisson equation*

$$(I - P)g + e\eta = f. \tag{1}$$

Its solution $g = (g(1), \cdots, g(M))^T$ is called a *performance potential* vector, and $g(i)$ is the potential at state i. g is also called the value function in dynamic programming, or the "differential" or "relative cost vector" [3], and "bias". The solution to (1) is only up to an additive constant; i.e., if g is a solution to (1), then so is $g + ce$.

Let P' and π' be another irreducible transition probability matrix on the same state space and its steady state probability. Let f' be the performance function for the system with P', $Q = P' - P = [q(i, j)]$ and $h = f' - f$. Then $Qe = 0$. The steady state performance corresponding to P' is $\eta' = \pi' f'$. Multiplying both sides of (1) with π', we can verify that

$$\eta' - \eta = \pi'(Qg + h). \tag{2}$$

Now, suppose that P changes to $P(\delta) = P + \delta Q = \delta P' + (1 - \delta)P$, and f changes to $f(\delta) = f + \delta h$, with $\delta \in (0, 1]$. Then the performance measure changes to $\eta(\delta) = \eta + \Delta\eta(\delta)$. The derivative of η in the direction of Q is defined as $\frac{d\eta}{d\delta} = \lim_{\delta\to 0} \frac{\Delta\eta(\delta)}{\delta}$. Taking $P(\delta)$ as the P' in (2), we have $\eta(\delta) - \eta = \pi(\delta)(\delta Qg + \delta h)$. Letting $\delta \to 0$, we get

$$\frac{d\eta}{d\delta} = \pi(Qg + h). \tag{3}$$

For references, see, e.g., [6], [7]. Since $Qe = 0$, for any g satisfying (1) for any constant c, we have $Qg = Q(g + ce)$, thus both (3) and (2) still hold for $g' = g + ce$. This verifies again that potentials are determined only up to an additive constant; this is the same as the potential energy in physics.

In (3), a linear structure $P(\delta) = P + \delta Q$ is assumed. In general, the transition probability matrix may depend on an arbitrary parameter θ, which is normalized in $[0, 1]$; i.e., $P(\theta) = P + Q(\theta)$ with $Q(1) = P(1) - P = P' - P$. Similarly, we assume $f(\theta) = f + h(\theta)$. Thus, for $\theta << 1$, we have $P(\theta) = P + \{\frac{dQ}{d\theta}\}_{\theta=0}\theta$, and $f(\theta) = f + \{\frac{dh}{d\theta}\}_{\theta=0}\theta$; i.e., in the neighboring area of $\theta = 0$, $P(\theta)$ and $f(\theta)$ take a linear form. Replacing Q in (3) with $\{\frac{dQ}{d\theta}\}_{\theta=0}$ and h with $\{\frac{dh}{d\theta}\}_{\theta=0}$ and noting that $\frac{dP}{d\theta} = \frac{dQ}{d\theta}$ and $\frac{df}{d\theta} = \frac{dh}{d\theta}$ we get

$$\frac{d\eta}{d\theta}|_{\theta=0} = \pi \left\{ (\frac{dP}{d\theta})_{\theta=0} g + (\frac{df}{d\theta})_{\theta=0} \right\}. \tag{4}$$

Therefore, without loss of generality, we shall mainly discuss the linear case (3).

The two simple equations (3) and (2) represent the performance sensitivity; (3) is the performance derivative (or gradient) with respect to continuous variables, and (2) is the performance difference for two discrete parameters (P and P'). Both of them depend mainly on the same quantity: the performance potential. Note that both depend on only potential g (not g'), and π and g can be estimated based on a single sample path of the Markov chain with transition matrix P (see section III-A).

The two equations (3) and (2) form the basis for performance optimization of Markov systems. Two basic approaches can be developed from them. The first one is the gradient-based optimization, which combines the gradient estimation based on (3) and the stochastic approximation techniques. This approach applies to systems that can be parameterized by continuous variables. This is in the same spirit as the perturbation analysis (or PA) based optimization (see, e.g. [9], [10]). The sensitivity formula (3) can indeed be derived by applying the PA principles. The second approach is the policy-iteration based optimization. It can be shown that policy iteration algorithms in Markov decision problems can be easily derived from (2) (see, e.g., [6]). The main issues here is to design fast policy iteration procedures that converge to the optimal policy. Both approaches depend heavily on the estimation of potentials. Q-learning [14], actor-critic type of algorithms, etc., are variants of this approach: they aim at to find directly the potentials (or the equivalent Q-factors) for the optimal policy. These are simulation based algorithms since they require the sample path to visit very state-action pair.

III. ESTIMATION OF POTENTIALS AND PERFORMANCE DERIVATIVES

A. Estimation of Performance Potentials

We first brief review that the potentials of a Markov chain can be estimated with a single sample path of the Markov chain. Since g is only up to an additive constant, we may choose the one that satisfies $\pi g = \pi f = \eta$. Thus, (1) becomes

$$(I - P + e\pi)g = f. \tag{5}$$

With (5), the potential g can be estimated on a sample path either "directly" [8], or by TD(λ) algorithms etc, [14].

B. Gradient Estimation

Consider a stationary Markov chain $\mathbf{X} = (X_0, X_1, \cdots,)$. (This implies the initial probability distribution is π.) Let E denote the expectation on the probability space generated by $\mathbf{X}$. Denote a generic time instant as k. Because it is impossible for a sample path with P to contain information about P', we need to use a standard technique in simulation called *importance sampling*. We have

$$\begin{aligned} \frac{d\eta}{d\delta} &= \pi(Qg + h) \\ &= \sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{S}} \{\pi(i)[p(i, j)\frac{q(i, j)}{p(i, j)} g(j) + h(i)]\} \end{aligned}$$

$$= E\left\{\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}g(X_{k+1})+h(X_k)\right\}. \quad (6)$$

Furthermore, if $\hat{g}$ is a random variable defined on $\mathbf{X}$ such that $E(\hat{g})=g$ and $\hat{g}$ is independent of the transition from X_k to X_{k+1}, then we have

$$\frac{d\eta}{d\delta}=E\left\{\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}\hat{g}+h(X_k)\right\}. \quad (7)$$

Sample path based algorithms can be developed by using (7) and the estimates of g. Let us first use

$$g_L(i)\approx E[\sum_{l=0}^{L-1}f(X_l)|X_0=i].$$

Each term in πQg takes the form $\pi(i)q(i,j)g(j)$. Note that (for notational simplicity, we set $h=0$)

$$\begin{aligned}&\pi(i)p(i,j)g(j)\\ =\;& E\{\epsilon_i(X_k)\epsilon_j(X_{k+1})g(X_{k+1})\}\\ \approx\;& \lim_{K\to\infty}\frac{1}{K-L+1}\times\\ &\left\{\sum_{k=0}^{K-L}\epsilon_i(X_k)\epsilon_j(X_{k+1})[\sum_{l=0}^{L-1}f(X_{k+l+1})]\right\}, w.p.1 \quad (8)\end{aligned}$$

The convergence can be proved. Defining a function $Z_k=\epsilon_i(X_k)\epsilon_j(X_{k+1})[\sum_{l=0}^{L-1}f(X_{k+l+1})]$, we get an ergodic process $Z=\{Z_k,k\geq 0\}$. Therefore, the right-hand side of (8) equals

$$\begin{aligned}&E\{\epsilon_i(X_k)\epsilon_j(X_{k+1})[\sum_{l=0}^{L-1}f(X_{k+l+1})]\}\\ =\;& E\{\sum_{l=0}^{L-1}f(X_{k+l+1})|\epsilon_i(X_k)\epsilon_j(X_{k+1})=1\}\times\\ &p^*[\epsilon_i(X_k)\epsilon_j(X_{k+1})=1],\end{aligned}$$

where p^* is the steady-state probability of $X_k=i$ and $X_{k+1}=j$. By the Markov property, the first term equals $g(j)$, and the second term equals $\pi(i)p(i,j)$.

From (8), we have [8]

$$\begin{aligned}&\pi(i)q(i,j)g(j)\\ \approx\;& \lim_{K\to\infty}\frac{1}{K-L+1}\left\{\sum_{k=0}^{K-L}\epsilon_i(X_k)\epsilon_j(X_{k+1})\times\right.\\ &\left.\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}[\sum_{l=0}^{L-1}f(X_{k+l+1})]\right\}, w.p.1.\end{aligned}$$

In the above, the quantity involving $q(i,j)$, $\pi(i)q(i,j)g(j)$, is estimated by simulating a quantity involving $p(i,j)$, $\pi(i)p(i,j)g(j)$. This is a variant of the standard importance sampling technique in simulation, which is widely applied to study the performance of a stochastic system with a probability distribution by simulating another stochastic system with a different probability distribution.

Finally, by the ergodicity, we have

$$\begin{aligned}&\frac{\partial\eta}{\partial\delta}\approx\pi Qg_L\\ =\;& \lim_{K\to\infty}\frac{1}{K-L+1}\sum_i\sum_j\left\{\sum_{k=0}^{K-L}\epsilon_i(X_k)\times\right.\\ &\left.\epsilon_j(X_{k+1})\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}[\sum_{l=0}^{L-1}f(X_{k+l+1})]\right\}\\ =\;& \lim_{K\to\infty}\frac{1}{K-L+1}\left\{\sum_{k=0}^{K-L}\sum_i\sum_j\{\epsilon_i(X_k)\times\right.\\ &\left.\epsilon_j(X_{k+1})\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}\}[\sum_{l=0}^{L-1}f(X_{k+l+1})]\right\}\\ =\;& \lim_{K\to\infty}\frac{1}{K-L+1}\times\\ &\left\{\sum_{k=0}^{K-L}\{\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}\}[\sum_{l=0}^{L-1}f(X_{k+l+1})]\right\}, w.p.1. \quad (9)\end{aligned}$$

It can be shown that (9) is equivalent to [8]

$$\begin{aligned}&\frac{\partial\eta}{\partial\delta}\approx\lim_{K\to\infty}\frac{1}{K-L+1}\times\\ &\sum_{k=0}^{K-L}\left\{f(X_{k+L})\sum_{l=0}^{L-1}\left[\frac{q(X_{k+l},X_{k+l+1})}{p(X_{k+l},X_{k+l+1})}\right]\right\}, w.p.1. \quad (10)\end{aligned}$$

In (9) and (10), g is approximated by truncation. We can also use an α-potential g_α, $0<\alpha<1$, to approximate g. g_α satisfies the following discounted Poisson equation:

$$(I-\alpha P+\alpha e\pi)g_\alpha=f.$$

It is shown that

$$\lim_{\alpha\to 1}g_\alpha=g.$$

Ignoring the constant term, we have

$$g_{\alpha,L}(i)=E[\sum_{l=0}^{\infty}\alpha^l f(X_l)|X_0=i].$$

Using this as the $\hat{g}$ in (7), we get (cf. (9))

$$\begin{aligned}&\frac{\partial\eta}{\partial\delta}\approx\lim_{K\to\infty}\frac{1}{K-L+1}\times\\ &\left\{\sum_{k=0}^{K-L}\{\frac{q(X_k,X_{k+1})}{p(X_k,X_{k+1})}\}[\sum_{l=0}^{\infty}\alpha^l f(X_{k+l+1})]\right\}, w.p.1. \quad (11)\end{aligned}$$

This is equivalent to (c.f. (10))

$$\begin{aligned}&\frac{\partial\eta}{\partial\delta}\approx\lim_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1}\times\\ &\left\{f(X_k)\sum_{l=0}^{k-1}\left[\alpha^{k-l-1}\frac{q(X_l,X_{l+1})}{p(X_l,X_{l+1})}\right]\right\}, w.p.1. \quad (12)\end{aligned}$$

An algorithm is developed in [1] to estimate $\frac{\partial\eta}{\partial\delta}$ using (12). It is easy to estimate $z_k:=\sum_{l=0}^{k-1}\left[\alpha^{k-l-1}\frac{q(X_l,X_{l+1})}{p(X_l,X_{l+1})}\right]$

recursively:

$$z_{k+1} = \alpha z_k + \frac{q(X_k, X_{k+1})}{p(X_k, X_{k+1})}.$$

On the other hand, to estimate $\sum_{l=0}^{L-1} \left[\frac{q(X_{k+l}, X_{k+l+1})}{p(X_{k+l}, X_{k+l+1})}\right]$, one has to store L values.

(7) can also be used to develop sample path based algorithms for the performance derivatives. We first choose any regenerative state i^*. For convenience, we set $X_0 = i^*$ and define $u_0 = 0$, and $u_{m+1} = min\{n : n > u_m, X_m = i\}$ be the sequence of regenerative points. Set $g(i^*) = 0$. For any $X_n = i \neq i^*$ and $u_m \leq n < u_{m+1}$ we have

$$g(X_n) = d(i^*, i) = E\{\sum_{l=n}^{t_{m+1}-1} [f(X_l) - \eta]\}.$$

With this and by (7), we have

$$\frac{d\eta}{d\delta} = E\left\{\frac{q(X_k, X_{k+1})}{p(X_k, X_{k+1})}\{\sum_{l=k+1}^{t_{m+1}-1} [f(X_l) - \eta]\} + h(X_k)\right\}. \quad (13)$$

Sample path based algorithms can then be developed, and we will not go into the details.

IV. GRADIENT-BASED OPTIMIZATION

Any gradient estimate (PA, LR or SF, or the potential based estimates discussed in section III) can be used together with the standard stochastic gradient algorithms for optimizing the cost in Markov decision processes. For applications of PA and LR to the optimization problems, see e.g., [10].

[13] proposed a potential-based recursive algorithm for optimizing the average cost in finite state Markov reward processes that depend on a set of parameters denoted as θ. The approach is based on the regenerative structure of a Markov chain. The gradient estimate is similar to (13) except that the performance η is also estimated on the sample path and the gradient is not estimated explicitly in each step of the recursion, because its estimate is used in the stochastic gradient algorithm to determine the step size in a recursive procedure to reach the value θ at which the performance gradient is zero. The paper also provides an elegant proof for the convergence of the algorithm.

The gradient based approach can be easily extended to the partially observable Markov decision processes (POMDP) (see, e.g., [1], [2]). The POMDP model in [1], [2] is described as follows. In addition to the state space $\mathcal{S} = \{1, \cdots, M\}$, there are a control space denoted as $\mathcal{U} = \{1, \cdots, N\}$ consisting of N controls and an observation space $\mathcal{Y} = \{1, \cdots, L\}$ consisting of L observations. Each $u \in \mathcal{U}$ determines a transition probability matrix P^u, which does not depend on the parameter θ. When the Markov chain is at state $i \in \mathcal{S}$, an observation $y \in \mathcal{Y}$ is obtained according to a probability distribution $\nu_i(y)$. For any observation y, we may choose a random policy $\mu_y(u)$, which is a probability distribution over the control space $\mathcal{U}$. It is assumed that the distribution depends on the parameter θ and therefore is denoted as $\mu_y(\theta, u)$.

Given an observation distribution $\nu_i(y)$ and a random policy $\mu_y(\theta, u)$, the corresponding transition probabilities are

$$p_\theta(i, j) = \sum_{u,y} \{\nu_i(y)\mu_y(\theta, u)p^u(i, j)\}.$$

Therefore,

$$\frac{d}{d\theta} p_\theta(i, j) = \sum_{u,y} \{\nu_i(y)p^u(i, j)\frac{d}{d\theta}\mu_y(\theta, u)\}. \quad (14)$$

In POMDP, we assume that although the state X_k, $k = 0, 1, \cdots$, is not completely observable, the cost $f(X_k)$ is known. Thus, algorithms can be developed by replacing $q(i, j)$ with $\frac{d}{d\theta} p_\theta(i, j)$ of (14) in the algorithms developed for standard MDPs in section III-B. For example, if $h(i) = 0$ then (13) becomes

$$\frac{d\eta}{d\delta} = E\left\{\frac{\frac{d}{d\theta} p_\theta(X_k, X_{k+1})}{p(X_k, X_{k+1})}\{\sum_{l=k+1}^{t_{m+1}-1} [f(X_l) - \eta]\}\right\}, \quad (15)$$

in which $f(X_k)$ is assumed to be observable.

A recursive algorithm called GPOMDP is presented in [1]. The algorithms uses a discount factor to approximate g (cf. (12)).

V. POLICY ITERATION

Omitted to reduce the length.

VI. CONCLUSION

We have provided an overview of learning and optimization from a system point of view. It provides a unified framework for PA, MDP, and RL. Many or most results reviewed here are not new; however, this new perspective does lead to some new research directions, such as the gradient-based policy iteration and the event based sensitivity analysis by the construction method. Further research is needed for these topics. We summarize the results by Figure 1.

VII. REFERENCES

[1] J. Baxter and P. L. Bartlett, "Infinite-Horizon Policy-Gradient Estimation," *Journal of Artificial Intelligence Research*, Vol. 15, 319-350, 2001.

[2] J. Baxter, P. L. Bartlett, and L. Weaver "Experiments with Infinite-Horizon Policy-Gradient Estimation," *Journal of Artificial Intelligence Research*, Vol. 15, 351-381, 2001.

[3] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, volume I and II. Athena Scientific, Belmont, MA, 1995.

[4] X. R. Cao, Convergence of Parameter Sensitivity Estimates in a Stochastic Experiment, *IEEE Trans. on Automatic Control*, Vol. AC- 30, 834-843, 1985.

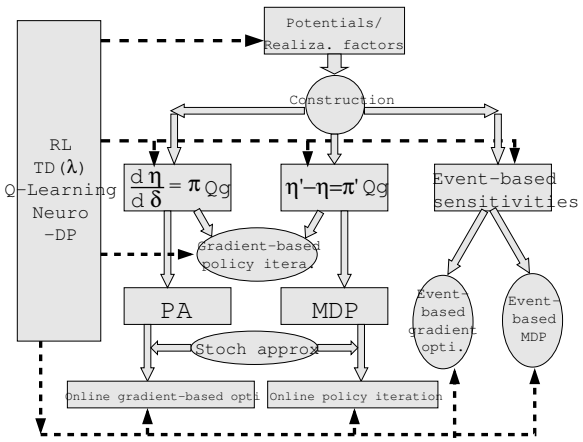

Fig. 1. A System Point of View of Learning and Optimization

[5] X. R. Cao, *Realization Probabilities: The Dynamics of Queueing Systems*, Springer-Verlag, New York, 1994.

[6] X. R. Cao, The Relation Among Potentials, Perturbation Analysis, Markov Decision Processes, and Other Topics, *Journal of Discrete Event Dynamic Systems*, Vol. 8, 71-87, 1998.

[7] X. R. Cao and H. F. Chen, Perturbation realization, potentials and sensitivity analysis of Markov processes, *IEEE Trans. on Automat. Control*, Vol. 42, 1382-1393, 1997.

[8] X. R. Cao and Y.W. Wan, Algorithms for sensitivity analysis of Markov systems through potentials and perturbation realization, *IEEE Trans. on Control System Tech*, Vol. 6, 482-494, 1998.

[9] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, 1999.

[10] E. K. P. Chong and P. J. Ramadge, Stochastic Optimization of Regenerative Systems Using Infinitesimal Perturbation Analysis, *IEEE Trans. on Automat. Control*, Vol. 39, 1400-1410, 1994.

[11] Hai-Tao Fang and X. R. Cao, Potential-Based Online Policy Iteration Algorithms for Markov Decision Processes, *IEEE Transactions on Automatic Control*, to appear.

[12] Y. C. Ho and X. R. Cao, *Perturbation Analysis of Discrete-Event Dynamic Systems*, Kluwer Academic Publisher, Boston, 1991.

[13] P. Marbach and T. N. Tsitsiklis, Simulation-based optimization of Markov reward processes, *IEEE Trans. on Automat. Control*, Vol. 46, 191-209, 2001.

[14] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, 1998.

[15] J. N. Tsitsiklis and B. Van Roy, Average Cost Temporal-Difference Learning, *Automatica*, Vol. 35, 1799-1808, 1999.