SCR-TECHNICAL REPORT -1995

Incremental Learning of Evaluation Functions for Absorbing Markov Chains: New Methods and Theorems

Leonid Gurvits
Long-Ji Lin
Stephen José Hanson

Siemens Corporate Research, Inc.
755 College Road East
Princeton, NJ 08540

November 8, 1994

Abstract

Consider playing a game such as chess against an opponent. At the end of the game we receive a reward, no reward, or a negative reward (penalty) depending on the final outcome, win, draw, or loss. Assuming a fixed strategy is employed to play the game, we want to learn to predict the expected reward starting from any board position. If we can make perfect predictions, then we can improve our playing strategy by taking the moves that lead to board positions having high expected rewards. This learning problem in fact belongs to a very general problem; that is, learning an evaluation function of states for absorbing Markov chains. There are two approaches to this general learning problem. The indirect approach learns a model of the underlying Markov chain and solves a system of linear equations, while the direct approach attempts to solve the problem without learning a model. The former is often infeasible except for small Markov chains. In the previous literature, many researchers have focused on a direct approach, the TD(λ) method. TD(λ) is incremental and efficient, and its convergence has been proved (with some small flaw, however). In this paper, we significantly extend the TD(λ) method, and present a general class of TD methods. We also fix some flaws in some previous convergence proofs, and provide a convergence proof for this new class of methods.

1 Introduction

A wide range of optimal control problems and sequential decision problems involve evaluating the goodness of system states. For example, playing chess is a sequential decision problem. At the end of each game, we receive a reward, no reward, or a negative reward (penalty) depending on the final outcome of the game, win, draw, or loss. Assuming a fixed (but possibly stochastic) strategy is now employed to play the game, if we can correctly predict the expected reward starting from any board position, then we can improve our playing strategy simply by taking the moves that lead to board positions having high expected rewards. Consider another problem, robot control. A robot has a set of primitive actions and a task to achieve. The robot has to learn to choose actions in order to accomplish the task optimally. The robot will receive a reward when one of the goal states is reached, and a penalty when the robot gets into an undesirable state such as collision. Once again, assuming a fixed strategy is now employed to choose actions, if the robot can predict the expected payoff starting from any system state, then it can improve its control strategy by choosing actions that lead to states with high expected payoffs.

The above two learning problems, game playing and robot control, are in fact instances of a general problem, learning an evaluation function of system states. In this paper, we only consider the case where states are discrete and completely observable. In other words, we consider problems that can be expressed as *Markov chains*. Formally, this general learning problem can be formulated as follows: Consider a Markov chain which has multiple absorbing states and non-absorbing states. Each absorbing state is associated with a scalar number called *terminal value* or *reinforcement signal*. The objective of learning is to predict the expected terminal value given that the system is now in any of the system states.

There are *direct* ways and *indirect* ways to learn the evaluation functions. An indirect way first learns a model of the underlying Markov chain (including the transition matrix and the terminal values), and then solves a system of linear equations (see Section 2). A direct way, on the other hand, attempts to find the evaluation function directly without learning a model. Indirect ways are often quite computationally expensive and thus infeasible except for small Markov chains. In the previous literature regarding the direct approach, researchers have mostly focused on the TD(λ) method, a simple, incremental, and computationally feasible, direct approach. The TD(λ) method was first defined by Sutton [9], although the fundamental idea behind TD(λ) has been around in the literature for years (e.g., [7]). Here λ is a parameter between 0 and 1 defining a specific algorithm, and TD stands for *temporal difference* due to that TD(λ) constructs an evaluation function by minimizing the discrepancy between the predictions for *temporally successive* states.

Sutton [9] proved the convergence of TD($\lambda = 0$), and later Dayan [2] proved the convergence of the method for general λ. However, both proofs are flawed, because Sutton and Dayan made the same mistake in their proofs. See Section 5 for the mistake and our fix. In [3], Dayan and Sejnowski prepare a "convergence with probability 1" proof for TD(λ). In their proof, they use a projection technique to bound the variance of errors. However, such a projection technique in general can destroy the convergence of TD(λ). In this paper, we provide a new proof which does not require the use of a projection technique. We also

provide the conditions under which a projection technique can be used without destroying the convergence property (Section 4.1).

As pointed out by Watkins [12] and Barto, et al [1], TD can be considered as a type of *asynchronous dynamic programming*. A successful application of the TD method is *Q-learning*, first defined by Watkins [12]. Q-learning is a direct method for optimal control. Its convergence was proved by Watkins. The convergence analyses of the TD(λ) method and Q-learning are all based on the use of look-up tables (or linear representations) to store evaluation functions. In practice, non-linear function approximators have been employed to solve some nontrivial learning problems. For example, Tesauro [10] developed a backgammon playing program which learned to play as well as world-class human players. Lin [5] also had a program which could learn to survive in a simulated environment filled with predators and food. Both programs used *artificial neural networks* as function approximators.

In this paper, we extend TD(λ). The results include (1) a general class of temporal difference methods called $TD(C_{tk})$, of which TD(λ) is a special case and (2) a convergence proof, which we have tried to make as simple and self-contained as possible. One of the goals of this paper is to correct some mathematical flaws of previous work, although the theoretical results of the previous work are in general correct. In the first part of this paper, we introduce the most general linear weight-update operator in a spirit of temporal differencing. As a first result, we obtain the recursive expression of the prediction errors which is expected from using TD(C_{tk}). We also give a few instances of this new learning method, including TD(λ). In the second part, we provide conditions under which TD(C_{tk}) can be proved to converge with probability one. Our proof is complete (and we hope, correct). In the third part, we provide necessary and sufficient conditions for $TD(C_{tk})$ to converge for any Markov chain. Finally, we discuss $TD(C_{tk})$ methods for predicting cumulative outcomes, and show its convergence.

2 Notations and Basic Facts

Figure 1 shows the notations we use in this paper. The evaluation function is represented by $V = X^T w$, where X is the state representation and w is a set of weights. The objective here is to obtain a set of weights w such that $X^T w$ approaches V^*, the true evaluation function. A straightforward state representation is that $X = I$. With this representation, the expected terminal value from state i is simply w_i. This representation can be easily implemented by a lookup table, which has an entry for each state i to store w_i.

The learning of the weight vector w is divided into *epochs*. At the start of each epoch, $t = 0$ and the system's initial state, $s(0)$, is determined by a probability distribution, u. The system's next states will then be governed by the transition matrix, T. Each epoch ends either when an absorbing state is reached or when some pre-defined time has expired. After each epoch, our proposed learning algorithm will update w based on the information contained in this epoch about the underlying Markov chain.

Note that $\{s(t)|t \geq 0\}$ stands for a *random walk* through the underlying Markov chain. A random walk is allowed to terminate before an absorbing state is reached. If a random walk is absorbed at time t (i.e., $j = s(t) \in \mathcal{A}$), then we define $s(t') = j$ for all $t' > t$. If a

$\{c_i\}$	column vector with component c_i
I	identity matrix
O	matrix with all entries 0's
$Diag(c)$	diagonal matrix whose diagonal is vector c
A^T	the transpose of A
$\sigma(A)$	the eigenvalues of A
$\rho(A)$	the largest eigenvalue (in modulus) of A
$< a, b >$	$a \cdot b$, the inner product of vectors a and b
$\mathcal{N}$	non-absorbing states or non-terminals
$\mathcal{A}$	absorbing states or terminals
T	Transition matrix $T = \left(\begin{array}{c\|c} I & O \\ \hline R & Q \end{array}\right)$
R	probabilities of transition from non-absorbing to absorbing states
Q	probabilities of transition between non-absorbing states
u_i	probability that the system starts at state $i \in \mathcal{N}$
u	$u = \{u_i \| i \in \mathcal{N}\}$, the *initial state distribution*
X_i	column vector representing state $i \in \mathcal{N}$
X	matrix whose columns are X_i, $i \in \mathcal{N}$
z_j	expected terminal value from state $j \in \mathcal{A}$
z	$z = \{z_j \| j \in \mathcal{A}\}$
t	time
$s(t)$	state at time t
w	column vector representing a set of weights
V_i	prediction of the expected terminal value starting from state $i \in \mathcal{N}$ $V_i = X_i^T \cdot w$
V	$V = \{V_i \| i \in \mathcal{N}\} = X^T w$, the estimated evaluation function
V^*	the true evaluation function
d_i	number of times in state $i \in \mathcal{N}$ before absorption during a walk on the Markov chain
d	$d = \{d_i \| i \in \mathcal{N}\}$
B_{ij}	probability starting in state $i \in \mathcal{N}$ that the Markov chain is absorbed in state $j \in \mathcal{A}$
B	matrix whose elements are B_{ij}

Figure 1: Notations.

1. $\rho(Q) < 1$; in other words, $lim_{n\to\infty} Q^n = O$.
2. $d = \{d_i\} = u + uQ + uQ^2 + uQ^3 + \cdots = u(I - Q)^{-1}$
3. $B = R + QR + Q^2R + Q^3R + \cdots = (I - Q)^{-1}R$
4. $R = (I - Q)B$
5. $$T^n = \left(\begin{array}{c|c} I & O \\ \hline (I + Q + Q^2 + \cdots + Q^{n-1})R & Q^n \end{array}\right)$$
6. $V^* = Bz = (I - Q)^{-1}Rz$

Figure 2: Facts about absorbing Markov chains.

random walk ends at time t without absorption (i.e., $i = s(t) \in \mathcal{N}$), then $s(t')$ is undefined for all $t' > t$.

Figure 2 gives a list of facts about absorbing Markov chains that will be used in this paper. Note that by Fact 6, the true values V^* can be directly computed if the transition matrix (T) and the expected terminal values from absorbing states (z) are known or learned. This direct computation, however, will be costly for large Markov chains. Moreover, in the case that learning has to take place on-line, this costly computation will need to be repeated every time when new information is available and the transition matrix gets adjusted, making the direct approach prohibitively costly.

3 TD(C_{tk}) : The Generalized TD Method

As mentioned before, the objective here is to obtain a weight vector w such that $X^T w \to V^*$. In this section, we develop a new generic learning method, TD(C_{tk}). As will be seen, the TD(λ) method can be easily put in our framework. In their pioneering work [9, 2], Sutton and Dayan obtained a recursive expression of the prediction errors which is expected from using TD(λ). However, their presentations are somewhat tangled. Here we present a clear and simple way to obtain the equivalent, but more general, recursive error expression for this very general learning method, TD(C_{tk}).

3.1 Weight-Update Operators

We consider an arbitrary Markov chain with a countable (only for the sake of simplicity) state space. Recall in Section 2 that we use T to denote the transition matrix of the Markov chain, and use u to denote the initial state distribution. We first consider a very simple operator, Δw, which uses a random walk (i.e., $\{s(t)|t \geq 0\}$) to update the weight vector w. In fact, only the first two states ($s(0)$ and $s(1)$) of the walk are utilized.

Definition 1

$$\Delta w \triangleq \begin{cases} (z_j - X_{s(0)}^T w) X_{s(0)} & \text{if } s(1) = j \in \mathcal{A} \\ (X_{s(1)}^T w - X_{s(0)}^T w) X_{s(0)} & \text{if } s(1) \in \mathcal{N} \end{cases} \quad (1)$$

To understand the meaning of this operator, let us consider the case that we use a lookup table to store our current estimate of the evaluation function (i.e., $X = I$); one table entry for each state. This Δw operator simply looks up the state values of the first two states in the random walk (i.e., $s(0)$ and $s(1)$), and then computes the difference. The following lemma gives what this difference is expected to be.

Lemma 1

$$\begin{aligned} \bigtriangledown(u,T) &\triangleq E(\Delta w \mid u \textit{ is the initial state distribution and } T \textit{ is the transition matrix}) \\ &= X\, Diag(u)\, (I - Q)(V^* - V) \end{aligned}$$

where $T = \left(\begin{array}{c|c} I & O \\ \hline R & Q \end{array}\right)$.

Proof:

$$\begin{aligned} & \bigtriangledown(u,T) \\ = & \sum_{i \in \mathcal{N}} u_i [\sum_{j \in \mathcal{N}} Q_{ij} (X_j^T w - X_i^T w) + \sum_{j \in \mathcal{A}} R_{ij} (z_j - X_i^T w)] X_i \\ = & \sum_{i \in \mathcal{N}} u_i [\sum_{j \in \mathcal{N}} Q_{ij} X_j^T w + \sum_{j \in \mathcal{A}} R_{ij} z_j] X_i - \sum_{i \in \mathcal{N}} u_i [\sum_{j \in \mathcal{N}} Q_{ij} + \sum_{j \in \mathcal{A}} R_{ij}] X_i^T w X_i \end{aligned}$$

Using the obvious fact that

$$\sum_{j \in \mathcal{N}} Q_{ij} + \sum_{j \in \mathcal{A}} R_{ij} = 1,$$

we obtain

$$\bigtriangledown(u,T) = X\, Diag(u)\, [QX^T w + Rz] \; - \; X\, Diag(u)\, X^T w.$$

From Facts 4 and 6, we further obtain

$$\begin{aligned} & \bigtriangledown(u,T) \\ = & \; X\, Diag(u)\, [(Q - I) X^T w + (I - Q) B z] \\ = & \; X\, Diag(u)\, [(Q - I) V + (I - Q) V^*] \\ = & \; X\, Diag(u)\, (I - Q)(V^* - V). \end{aligned}$$

□

Now we define a more general operator, Δw_{tk}, for weight update. This operator involves only two states in the given random walk, $s(t)$ and $s(t+k)$.

Definition 2

$$\Delta w_{tk} \triangleq \begin{cases} 0 & \text{if } s(t) \in \mathcal{A} \\ (z_j - X_{s(t)}^T w) X_{s(t)} & \text{if } s(t) \in \mathcal{N} \text{ and } s(t+k) = j \in \mathcal{A} \\ (X_{s(t+k)}^T w - X_{s(t)}^T w) X_{s(t)} & \text{if } s(t) \in \mathcal{N} \text{ and } s(t+k) \in \mathcal{N} \end{cases} \quad (2)$$

provided that $k > 0$ *and that* $s(t)$ *and* $s(t+k)$ *are defined.*

Lemma 2

$$\nabla_{tk}(u,T) \triangleq E(\Delta w_{tk} \mid u,T) = X\ Diag(uQ^t)\,(I-Q^k)(V^*-V) \tag{3}$$

Proof: Recall that $s(t+k)$ is undefined only when the random walk was terminated without absorption before time $t+k$. Let v be the probability distribution of the system being in a non-absorbing state at time t. Then $v = uQ^t$ Let M be the probability distribution of the system being absorbed to a terminal state at or before time $t+k$, provided that the system is not absorbed at time t. Then $M = (I+Q+Q^2+\cdots+Q^{k-1})R = (I-Q^k)(I-Q)^{-1}R$. This proof (for Lemma 2) is similar to that for Lemma 1. We simply replace u and $T = \left(\begin{array}{c|c} I & O \\ \hline R & Q \end{array}\right)$ used in that proof by v and $T^k = \left(\begin{array}{c|c} I & O \\ \hline M & Q^k \end{array}\right)$ in this proof:

$$\begin{aligned}
& \nabla_{tk}(u,T) \\
= & \sum_{i\in\mathcal{N}} v_i[\sum_{j\in\mathcal{N}} Q^k_{ij}(X_j^T w - X_i^T w) + \sum_{j\in\mathcal{A}} M_{ij}(z_j - X_i^T w)]X_i \\
= & X\ Diag(v)\,[Q^k X^T w + Mz - X^T w] \\
= & X\ Diag(v)\,[(Q^k - I)X^T w + (I-Q^k)Bz] \\
= & X\ Diag(uQ^t)\,(I-Q^k)(V^*-V)
\end{aligned}$$

□

Finally we define the most general update operator, $\Delta w(C_{tk})$, which involves every pair of states in the given random walk.

Definition 3

$$\Delta w(C_{tk}) \triangleq \sum_{t\geq 0}\sum_{k\geq 1} C_{tk}\Delta w_{tk} \tag{4}$$

where C_{tk} is a (possibly negative) pre-defined number for every $t \geq 0$ and every $k \geq 1$.

Theorem 1

$$E(\Delta w(C_{tk})|u,T) = X\sum_{t\geq 0}\sum_{k\geq 1} C_{tk}\ Diag(uQ^t)\,(I-Q^k)(V^*-V) \tag{5}$$

Proof: This theorem directly follows Lemma 2, because of the additiveness of mathematical expectation. □

Remark: Because of the condition in Definition 2, to use the $\Delta w(C_{tk})$ operator, we require that each random walk either ends in an absorbing state or has a length greater than or equal to l where $C_{tk} = 0$ for all $t+k > l$.

3.2 The TD(C_{tk}) Method and Its Error Dynamics

Method: Let w_n be the set of weights at epoch n. A new set of weights is obtained by the following update procedure:

$$w_{n+1} \leftarrow w_n + Diag(\alpha_n)\Delta w(C_{tk}) \tag{6}$$

where $\Delta w(C_{tk})$ is as defined in Definition 3, and α_n is a vector of learning rates used at epoch n.

Figure 3: The generalized TD method: TD(C_{tk}).

Figure 3 gives the generalized temporal difference (TD) method, TD(C_{tk}). This method simply updates the weight vector w by the amount of $\Delta w(C_{tk})$ multiplied by some learning rates α_n. We allow each state to have its own learning rate. We also allow the learning rates to be different at each epoch. For instance, we may anneal learning rates over time to obtain convergence of the weights. The method given in Figure 3 in fact defines a class of learning algorithms; each possible set of C_{tk} values defines a specific learning algorithm. Some algorithms may have nice convergence properties, while the others may not. For example, the examples given in Section 3.3 all have nice convergence properties.

Let e_n be the error between the truth V^* and the prediction $V = X^T w_n$ at epoch n. In other words, $e_n = V^* - V$ at epoch n. The following corollary describes the relationship between the expected e_{n+1} and the expected e_n.

Corollary 1 *Let e_n be the prediction error at epoch n (i.e., $e_n = V^* - X^T w_n$). The TD(C_{tk}) method given in Figure 3 has the following dynamic of prediction errors:*

$$E(e_{n+1}) = [I - X^T Diag(\alpha_n) X \sum_{t\geq 0} \sum_{k\geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k)] E(e_n) \tag{7}$$

Proof: From Theorem 1, we obtain

$$\begin{aligned}
& E(e_{n+1}) \\
= \; & E(V^* - X^T w_{n+1}) \\
= \; & E(V^* - X^T (w_n + Diag(\alpha_n)\Delta w(C_{tk}))) \\
= \; & E(V^* - X^T w_n) - X^T Diag(\alpha_n) E(\Delta w(C_{tk})) \\
= \; & E(V^* - X^T w_n) - X^T Diag(\alpha_n) X \sum_{t\geq 0} \sum_{k\geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k) E(V^* - X^T w_n) \\
= \; & [I - X^T Diag(\alpha_n) X \sum_{t\geq 0} \sum_{k\geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k)] E(e_n)
\end{aligned}$$

□

3.3 Instances of TD(C_{tk})

In this subsection, we show that TD(λ) can be easily put in our framework, TD(C_{tk}). In addition to TD(λ), we present a few more instances of TD(C_{tk}). All of them have nice

convergence properties, as will be proved in Section 5.

3.3.1 TD(λ)

TD(λ) [9] (Section 1) is a special case of TD(C_{tk}) when the following learning parameters are used:

- $Diag(\alpha_n) = \alpha I$ where α is a constant. In other words, a global learning rate α is used at any time and for all weight updates.
- $C_{tk} = (1-\lambda)\lambda^{k-1}$ for all t, where $-1 < \lambda \leq 1$. [1] There are two special cases:
  - $\lambda = 0$: For all t, $C_{tk} = 1$ for $k = 1$, and $C_{tk} = 0$ for $k > 1$.
  - $\lambda = 1$: For all t, $C_{tk} = 1$ for $k = \infty$, and $C_{tk} = 0$ for finite k.

One good thing about TD(λ) is that there is a very efficient way to implement this learning method (for example, [6]). The time complexity of weight updating at each epoch is merely $O(n)$ where n is the length of the random walk.

Note that C_{tk} as defined above is non-zero for all t and some k, except when $\lambda = 0$. Recalling the remark following Theorem 1, to apply TD(λ), except TD(0), requires each random walk to be absorbed without premature termination.

Replacing the above learning parameters in Corollary 1, we obtain:

$$\begin{aligned}
& E(e_{n+1}) \\
= \; & [I - \alpha X^T X \sum_{t\geq 0}\sum_{k\geq 1}(1-\lambda)\lambda^{k-1}\, Diag(uQ^t)\,(I-Q^k)]\, E(e_n) \\
= \; & [I - \alpha X^T X Diag(\sum_{t\geq 0} uQ^t)(1-\lambda)(\sum_{k\geq 1}\lambda^{k-1} - \sum_{k\geq 1}\lambda^{k-1}Q^k)]\, E(e_n) \\
= \; & [I - \alpha X^T X Diag(u(I-Q)^{-1})(1-\lambda)((1-\lambda)^{-1} - Q(1-\lambda Q)^{-1})\, E(e_n) \\
= \; & [I - \alpha X^T X Diag(u(I-Q)^{-1})(I-Q)(1-\lambda Q)^{-1}]\, E(e_n)
\end{aligned}$$

This error dynamic is exactly the same as the one derived by Dayan [2].

3.3.2 TD(λ,μ)

TD(λ,μ) is a special case of TD(C_{tk}) when a global constant learning rate α is used and

$$C_{tk} = \begin{cases} \frac{\lambda^k - \mu^k - \lambda^{k+1} + \mu^{k+1}}{\lambda - \mu} & \text{if } \lambda \neq \mu \\ k\lambda^{k-1} - (k+1)\lambda^k & \text{if } \lambda = \mu \end{cases}$$

where $0 \leq \lambda \leq 1$ and $0 \leq \mu \leq 1$. Like TD(λ), there is a very simple and efficient implementation for TD(λ,μ); its time complexity is also $O(n)$ where n is the length of the random walk. Note that TD(λ,μ) is equivalent to TD(μ,λ), and that TD(λ,μ) reduces to TD(λ) when $\mu = 0$ and $0 \leq \lambda \leq 1$. The error dynamic of TD(λ,μ) is:

$$E(e_{n+1}) = [I - \alpha X^T X Diag(u(I-Q)^{-1})(I-Q)(1-\lambda Q)^{-1}(1-\mu Q)^{-1}]\, E(e_n).$$

[1] The original TD(λ) as defined by Sutton does not allow negative λ. In Section 5, we prove its convergence for any $\lambda \in (-1, 1]$.

3.3.3 TD(C_k)

A special case of TD(C_{tk}) is when $C_{tk} = C_k \geq 0$ for all t and k, and $\sum_{k\geq 1} C_k = 1$. We call this special case TD(C_k). When a global constant learning rate α is used, the error dynamic is:

$$E(e_{n+1}) = [I - \alpha X^T X Diag(u(I-Q)^{-1}) \sum_{k\geq 1} C_k(I - Q^k)]\, E(e_n). \tag{8}$$

Note that TD(λ) is a special case of TD(C_k) when $0 \leq \lambda \leq 1$.

3.3.4 An Unbiased Learning Method

We now present a simple *unbiased* learning method, by which we mean the expected prediction error is 0 at any time from the beginning of learning: $E(V^* - X^T w_n) = 0$ for all $n > 0$. To obtain unbiasness, we need:

$$h = \sum_{t=0}^{N} uQ^t > 0 \tag{9}$$

for some number N. Here h is a vector whose elements indicate the expected number of times each of the non-absorbing states will be visited during the time interval $[0, N]$. The condition $h > 0$ means that there exits a number N which is large enough so that every state has non-zero probability of being visited in the time interval $[0, N]$. Note that $h > 0$ is obviously satisfied when $u > 0$.

Now we consider the characteristic polynomial of the transition matrix Q:

$$p(\lambda) = \det(\lambda I - Q) = a_n\lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda + a_0$$

where n is the number of non-absorbing states. Since $\rho(Q) < 1$, $p(1) \neq 0$. Also $p(Q) = 0$ by Caley-Hamilton Theorem.

Proposition 1 *When*

$$X = I,$$

$$Diag(\alpha_n) = (Diag(\sum_{t=0}^{N} uQ^t))^{-1}, \textit{and}$$

$$C_{tk} = \begin{cases} a_k/p(1) & \textit{if } 1 \leq k \leq n \textit{ and } t \leq N \\ 0 & \textit{otherwise,} \end{cases}$$

we obtain

$$E(V^* - X^T w_n) = 0 \qquad \textit{for all } n > 0$$

Proof: Using

$$p(1) = \sum_{k=0}^{n} a_k = p(1) \sum_{k=1}^{n} C_{tk} + a_0 \qquad \textit{and}$$

$$p(Q) = 0 = \sum_{k=0}^{n} a_k Q^k = p(1) \sum_{k=1}^{n} C_{tk} Q^k + a_0 I,$$

we obtain

$$\begin{aligned} & Diag(\alpha_n)X^TX\sum_{t\geq 0}\sum_{k\geq 1}C_{tk}\,Diag(uQ^t)\,(I-Q^k) \\ = \;& Diag(\alpha_n)Diag(\sum_{t=0}^{N}uQ^t)\sum_{k=1}^{n}C_{tk}\,(I-Q^k) \\ = \;& (\sum_{k=1}^{n}C_{tk})I-\sum_{k=1}^{n}C_{tk}Q^k \\ = \;& (1-\frac{a_0}{p(1)})I+\frac{a_0}{p(1)}I \\ = \;& I. \end{aligned}$$

By Corollary 1, $E(e_n)=0$. □

While interesting, this unbiased method may not be very practical. First, it requires to know the characteristic polynomial of Q (i.e., a_k) beforehand. Second, although the expected prediction error is 0 from the beginning of learning, the second moment of the prediction error (i.e., square of the error) may be large, making this unbiased method take a long time to converge.

4 Convergence with Probability One

In this section, we give a sufficient condition for TD(C_{tk}) to converge with probability one. Without loss of generality and for the sake of simplicity, below we mostly consider the case where $X=I$. When $X=I$, $V=w$.

Lemma 3 *In any finite Markov chain, there exist numbers $b>0$ and $0<c<1$ such that the probability that the system is still not absorbed after n steps is*

$$prob(n)\leq b\cdot c^n.$$

Proof: Let p_n be the probability of not reaching an absorbing state in n steps. Clearly, $p_n=(uQ^n)\cdot e$, where $e=(1,1,\cdots,1)$. Hence, $p_n\leq \|uQ^n\|_{L_1}\|e\|_{L_\infty}=\|uQ^n\|_{L_1}\leq\|Q^n\|\leq k(\rho(Q)+\varepsilon)^n$, where k is some constant and ε is an arbitrarily small positive number. Since $\rho(Q)<1$, the above lemma follows. □

Corollary 2 *The prediction errors of the TD(C_{tk}) method can be expressed recursively as follows:*

$$e_{n+1}=V^*-w_{n+1}=(I-Diag(\alpha_n)A_n)e_n+Diag(\alpha_n)B_n \tag{10}$$

where A_n is a random matrix, B_n is a random vector. Moreover

$$E(B_n)=0 \tag{11}$$

and

$$E(A_n)=\bar{A}=\sum_{t\geq 0}\sum_{k\geq 1}C_{tk}\,Diag(uQ^t)\,(I-Q^k) \tag{12}$$

Proof: Suppose the random walk at epoch n consists of N steps before absorption. Then

$$\Delta w(C_{tk}) = \sum_{t<N} \sum_{t+k<N} C_{tk}(w_n(s(t+k)) - w_n(s(t)))I_{s(t)} + \sum_{t<N} \sum_{t+k\geq N} C_{tk}(z - w_n(s(t)))I_{s(t)}$$

where $w_n(i)$ is the ith element of the weight vector w_n, I_i is a vector whose elements are 0 except for the ith element which is 1, and z is the terminal value received at the end of this random walk. Clearly, the above can be re-written as

$$\Delta w(C_{tk}) = Mw_n + zv$$

where matrix M and vector v are composed entirely of $\pm C_{tk}$. Thus we obtain

$$\begin{aligned} e_{n+1} &= V^* - w_{n+1} \\ &= V^* - (w_n + Diag(\alpha_n)(Mw_n + zv)) \\ &= (I + Diag(\alpha_n)M)(V^* - w_n) - Diag(\alpha_n)(MV^* + zv) \\ &= (I - Diag(\alpha_n)A_n)e_n + Diag(\alpha_n)B_n \end{aligned}$$

where

$$\begin{aligned} A_n &= -M \\ B_n &= -(MV^* + zv) \end{aligned}$$

An argument similar to the above also holds when the random walk terminates without absorption. Note that $E(A_n) = \bar{A}$ is basically the main results of Corollary 1, which also states that $E(e_{n+1}) = (I - Diag(\alpha_n)\bar{A})E(e_n)$. To prove that $E(B_n) = 0$, let $e_n = 0$, then we have that $E(e_{n+1}) = 0$ and that $E(e_{n+1}) = Diag(\alpha_n)E(B_n)$. Thus, $E(B_n) = 0$ for any n. □

Remark: A_n is a random matrix, and B_n is a random vector. Obviously, A_n is not independent of B_n. (A_i, B_i) and (A_j, B_j), however, are independent for all $i \neq j$, since we assume random walks are taken independently.

Proposition 2 *If*

$$\sum_{t<N} \sum_{k\geq 1} |C_{tk}| = O(N) \tag{13}$$

where N is any constant, then $\|A_n\|$ and $\|B_n\|$ have exponential tails. In other words, for any number $r > 0$, there exist numbers $b_1 > 0$, $b_2 > 0$, $0 < c_1 < 1$, and $0 < c_2 < 1$ such that

$$prob(\|A_n\| > r) \leq b_1 c_1^r \tag{14}$$

and

$$prob(\|B_n\| > r) \leq b_2 c_2^r \tag{15}$$

for any number $r > 0$ and some numbers b_1, b_2, c_1, and c_2 are some positive constants depending on Q. Consequently,

$$E(\|A_n\|^2) < \infty \tag{16}$$

and

$$E(\|B_n\|^2) < \infty. \tag{17}$$

Proof: Suppose the random walk at epoch n consists of N steps before absorption. In the proof for Corollary 2, we have shown that $A_n = -M$ and $B_n = -(MV^* + zv)$, where matrix M and vector v are composed entirely of $\pm C_{tk}$ for $t < N$. Moreover, each different C_{tk} appears at most two times in M and at most one time in v. Also note that for any arbitrary 2-D matrix H, $\|H\| \leq \sum_i \sum_j |H_{i,j}|$. Therefore if the random walk takes N steps before absorption, then

$$s(N) \stackrel{\Delta}{=} 2 \sum_{t<N} \sum_{k \geq 1} |C_{tk}| \geq \|A_n\|.$$

In other words, $prob(\|A_n\| > s(N))$ is not greater than the probability that the Markov system is not in an absorbing state after N steps, which we denote by $prob(N)$:

$$prob(\|A_n\| > s(N)) \leq prob(N).$$

From Lemma 3, we thus have

$$prob(\|A_n\| > s(N)) \leq b \cdot c^N$$

for some numbers $b > 0$ and $0 < c < 1$. From (4) and the assumption (13), Inequality (14) is obvious. Similarly we can prove Inequality (15) by further noting that both V^* and z are finite. From (14), (15), and simple calculus, it is easy to show (17) and (17).

□

Remark: The condition $\sum_{t<N} \sum_{k \geq 1} |C_{tk}| = O(N)$ is obviously satisfied if, for example, $C_{tk} = C_k$ for all t and $\sum_{k \geq 1} |C_k|$ is bounded. TD(λ), TD(λ,μ), and TD(C_k) all satisfy this condition.

Let us define a few new operators, which will be used in the rest of the paper.

Definition 4 (P-inner product)

$$\begin{aligned} < x, y >_A &\stackrel{\Delta}{=} < Ax, y > \\ \|x\|_A^2 &\stackrel{\Delta}{=} < x, x >_A \end{aligned}$$

Here x and y are vectors, A is a matrix, and $<\cdot,\cdot>$ is the usual inner product operator.

Definition 5 (P-adjoint operator)

$$< x, A_P^* y >_P \stackrel{\Delta}{=} < Ax, y >_P$$

where x and y are (complex) vectors, and A and P are matrices. It is easy to see that

$$A_P^* = P^{-1} A^T P.$$

Definition 6 (P-positive definite) *We call operator A* P-positive definite *(or $A >_P 0$) if $< Ax, x >_P > 0$. Obviously, $A >_P 0$ iff PA is positive definite in the usual sense.*

Later we will use the following equivalence:

$$PA + A^T P > 0 \Longleftrightarrow A + A_P^* > 0. \tag{18}$$

Also we will use below the following norm operator:

Definition 7 (Norm)

$$\|A\|_P \triangleq Max_{\|x\|_P=1} \|Ax\|_P$$

It is clear that

$$\|A\|_P^2 = \rho(AA_P^*) = \rho(AP^{-1}A^TP). \tag{19}$$

To further simplify our discussion, but without loss of generality, we will consider below only the case where we use $X = I$ and a global learning rate which is annealed over time:

$$Diag(\alpha_n) = \frac{\alpha}{n} I$$

where where α is some positive constant and n is the number of epochs attempted so far. The theorems and proofs to be presented below can be easily extended to deal with the cases where different weights can be updated with different learning rates and/or X is an arbitrary non-singular matrix.

Theorem 2 *Consider the following stochastic difference equation:*

$$e_{n+1} = (I - \frac{\alpha}{n+1} A_n)e_n + \frac{\alpha}{n+1} B_n. \tag{20}$$

As $n \rightarrow \infty$, $e_n \rightarrow 0$ almost sure (with probability 1), if the following conditions hold:

1. *α is positive,*
2. *(A_i, B_i) and (A_j, B_j) are independent for all $i \neq j$,*
3. *$E(B_n) = 0$,*
4. *$E(\|A_n\|^2) < \infty$,*
5. *$E(\|B_n\|^2) < \infty$, and*
6. *$E(A_n) = \bar{A}$ is an anti-stable matrix (i.e., $\sigma(\bar{A}) \subset \{z : Re(z) > 0\}$).*

Proof: Since all eigenvalues of $\bar{A}$ have positive real parts, by Lyapunov's Theorem [13], there exist a positive definite matrix $G > 0$ and some positive number g such that

$$G\bar{A} + \bar{A}^T G \geq gI.$$

Using the operators defined above, we obtain

$$\bar{A} + \bar{A}_G^* \geq \bar{g} I$$

for some $\bar{g} > 0$. Let us define *past* to be the history before Epoch $(n+1)$; that is, *past* $= (A_1, B_1, A_2, B_2, \cdots, A_n, B_n)$. Also let $\varepsilon = \frac{\alpha}{n+1}$, and $A_n = \bar{A} + \tilde{A}_n$, where $E(\tilde{A}_n) = 0$. Then

$$\begin{aligned} & E(\|e_{n+1}\|_G^2 \mid past) \qquad (21) \\ = \; & E(\|(I - \varepsilon A_n)e_n + \varepsilon B_n\|_G^2 \mid past) \\ = \; & E(< (I - \varepsilon A_n)e_n, (I - \varepsilon A_n)e_n >_G + 2 < (I - \varepsilon A_n)e_n, \varepsilon B_n >_G + \varepsilon^2 < B_n, B_n >_G \mid past) \\ = \; & E(\|(I - \varepsilon \bar{A})e_n\|_G^2) + \varepsilon^2 E(\|\tilde{A}_n e_n\|_G^2) - 2\varepsilon^2 < \tilde{A}_n e_n, B_n >_G + \varepsilon^2 E(\|B_n\|_G^2) \end{aligned}$$

$$\|I - \varepsilon\bar{A}\|_G^2 = \rho((I - \varepsilon\bar{A})(I - \varepsilon\bar{A})_G^*) = \rho(I - \varepsilon(\bar{A} + \bar{A}_G^*) + \varepsilon^2\bar{A}\bar{A}_G^*) \leq 1 - \varepsilon\bar{g} + \varepsilon^2\|\bar{A}\|_G^2$$

For sufficiently small ε, we have:

$$\|I - \varepsilon\bar{A}\|_G^2 \leq (1 - \varepsilon\bar{g}') \tag{22}$$

where $0 < \bar{g}' \leq \bar{g}$. Note also that if $E(\|A_n\|^2) < \infty$ and $E(\|B_n\|^2) < \infty$, then there exists a positive number k such that $E(\|A_n\|_G^2) \leq k < \infty$ and $E(\|B_n\|_G^2) \leq k < \infty$. Also, $| < \tilde{A}_n e_n, B_n > |_G \leq \sqrt{(\|\tilde{A}_n e_n\|_G^2)}\sqrt{(\|B_n\|_G^2)} \leq k\|e_n\|_G$. From these facts, (22), and (22), we obtain:

$$\begin{aligned} & E(\|e_{n+1}\|_G^2 \mid past) \\ \leq\ & (1 - \varepsilon\bar{g}')^2\|e_n\|_G^2 + \varepsilon^2 k\|e_n\|_G^2 + 2\varepsilon^2 k\|e_n\|_G + \varepsilon^2 k \\ \leq\ & (1 - \frac{\hat{g}}{n+1})^2\|e_n\|_G^2 + \frac{\hat{k}}{(n+1)^2}\|e_n\|_G + \frac{\hat{k}}{(n+1)^2} \end{aligned} \tag{23}$$

where $\hat{g}$ and $\hat{k}$ are some positive numbers. To get (24), we have assumed that $\varepsilon = \frac{\alpha}{n+1}$ is sufficiently small, which is true when n is large enough. Now let us consider two possible cases. In the first case: $\|e_n\|_G^2 \geq 1$. In this case, $\|e_n\|_G \leq \|e_n\|_G^2$, and

$$E(\|e_{n+1}\|_G^2 \mid past) \leq (\frac{n+1-\hat{g}}{n+1})^2\|e_n\|_G^2 + \frac{\hat{k}}{(n+1)^2}\|e_{n+1}\|_G + \frac{\hat{k}}{(n+1)^2} = C_1.$$

In the second case: $0 \leq \|e_n\|_G^2 \leq 1$. In this case, $\|e_n\|_G \leq 1$, and

$$E(\|e_{n+1}\|_G^2 \mid past) \leq (\frac{n+1-\hat{g}}{n+1})^2\|e_n\|_G^2 + \frac{\hat{k}}{(n+1)^2} + \frac{\hat{k}}{(n+1)^2} = C_2.$$

In any case,

$$\begin{aligned} E(\|e_{n+1}\|_G^2 \mid past)\ & \leq\ max(C_1, C_2) \\ \leq\ & (\frac{(n+1-\hat{g})^2 + \hat{k}}{(n+1)^2})\|e_n\|_G^2 + \frac{2\hat{k}}{(n+1)^2} \end{aligned}$$

It is clear that for sufficiently large n, $(n+1-\hat{g})^2 + \hat{k} \leq (n+1-\gamma)^2$, for some γ, $0 < \gamma < \hat{g}$. So $E(\|e_{n+1}\|_G^2 \mid past) \leq (1 - \frac{\gamma}{n+1})^2\|e_n\|_G^2 + \frac{2\hat{k}}{(n+1)^2}$, and of course, $E(\|e_{n+1}\|_G^2) \leq (1 - \frac{\gamma}{n+1})^2 E(\|e_n\|_G^2) + \frac{2\hat{k}}{(n+1)^2}$. Since $\prod_{n>0}(1 - \frac{\gamma}{n+1})^2 \to 0$ and $\sum_{n>0}\frac{2\hat{k}}{(n+1)^2} < \infty$, we conclude that $E(\|e_n\|_G^2) \to 0$ as $n \to \infty$.

To prove convergence with probability 1, we use standard supermartingale [8] idea. Since $\|e_n\|_G^2 \geq 0$, $E(\|e_{n+1}\|_G^2 \mid past) \leq \|e_n\|_G^2 + \frac{2\hat{k}}{(n+1)^2}$. Let us intoduce a new variable:

$$S_n = \|e_n\|_G^2 + \sum_{i \leq n-1}\frac{2\hat{k}}{(i+1)^2}.$$

Then $E(S_{n+1} \mid past) \leq S_n$, where $S_n \geq 0$. $S_n \to \hat{S}$ with probability 1, for some $\hat{S}$. But $S_n = \|e_n\|_G^2 + L_n$, where L_n approaches some number $\hat{L}$ as n approaches infinity. Therefore,

$\|e_{n+1}\|_G^2$ also approaches some (random) x with probability 1. But since we have already proved that $E(\|e_n\|_G^2) \rightarrow 0$, $\bar{x} = 0$ almost sure. □

From Corollary 2, Proposition 2, and Theorem 2, we obtain the following theorem:

Theorem 3 *With the following learning parameters:*

$$X = I,$$

$$\sum_{t<N} \sum_{k \geq 1} |C_{tk}| = O(N),$$

and

$$Diag(\alpha_n) = \frac{\alpha}{n} I,$$

the TD(C_{tk}) method depicted in Figure 3 guarantees that the prediction error e_n converges to 0 with probability 1 (as the number of epochs $n \rightarrow \infty$), provided that the learning rate α is positive and that

$$\bar{A} = \sum_{t \geq 0} \sum_{k \geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k)$$

is an anti-stable matrix.

Theorem 3 can be made more general. We still have the "convergence with probability 1" property even when $\sum_{t<N} \sum_{k \geq 1} |C_{tk}| = O(N^d)$ where $d \geq 1$ is finite. More generally, in the case that X is any arbitrary non-singular matrix and

$$Diag(\alpha_n) = \frac{1}{n} Diag(\alpha)$$

where α is a vector of learning rates, we also have "convergence with probability 1" provided that the following $\bar{A}$ is anti-stable:

$$\bar{A} = X^T Diag(\alpha) X \sum_{t \geq 0} \sum_{k \geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k).$$

4.1 Projection

Suppose V_i^* is bounded between a and b for all i. One may want to use a projection operator to force V_i to have values between a and b each time after the TD operator is applied.

Corollary 3 *If $e_{n+1} = F_n((I - \frac{\alpha}{n+1} A_n)e_n + \frac{\alpha}{n+1} B_n)$ and F_n is a contraction to 0 in the sense of $\| \cdot \|_G$, then $e_n \rightarrow 0$ with probability 1.*

Suppose that G is a diagonal matrix. Consider the following "saturation" mapping:

$$F_{(a_1,b_1),(a_1,b_1),\cdots,(a_n,b_n)}(y_1, y_2, \cdots, y_n) = (z_1, z_2, \cdots, z_n)$$

where

$$z_i = \left\{ \begin{array}{ll} a_i & \text{if } y_i \leq a_i \\ b_i & \text{if } y_i \geq b_i \\ y_i & \text{if } a_i \leq y_i \leq b_i. \end{array} \right).$$

If $x \in [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_n, b_n]$, $\|F(y) - x\|_G \leq \|y - x\|_G$. In our problem, if $X^T Diag X$ is diagonal and if we know that $\alpha \leq V_i \leq \beta$, then we can, after each iteration, project V_i onto $[\alpha, \beta]$. But in general, this kind of projection can prevent TD methods from convergence. Consider the simplest case. $e_{n+1} = Sat((I - \frac{1}{n+1}A)e_n)$, where

$$Sat(x) = \left\{ \begin{array}{ll} 1 & \text{if } x \geq 1 \\ x & \text{if } -1 \leq x \leq 1 \\ -1 & \text{if } x \leq -1. \end{array} \right).$$

We construct a matrix A such that all eigenvalues of A have positive real parts, and $A(1,1)^T = (-\alpha, -\beta)$, where $\alpha > 0$, $\beta > 0$, and $\alpha \neq \beta$. Then it is clear that if $e_0 = (1,1)$, e_n does not converge to 0. (Notice that without the saturation mapping, $e_n \rightarrow 0$.) Below we present an example of such a matrix A: $A = \begin{pmatrix} 2 & -8 \\ 0.5 & -1 \end{pmatrix}$. Indeed, $tr(A) = 1 > 0$, and $det(A) = 2 > 0$.

5 Convergence of TD(λ), TD(λ, μ), and TD(C_k)

Roughly speaking, so far we have obtained the following main result: TD(C_{tk}) converges with probability 1 if

1.
$$\bar{A} = X^T X \sum_{t \geq 0} \sum_{k \geq 1} C_{tk}\, Diag(uQ^t)\,(I - Q^k) \tag{24}$$

is anti-stable, and

2.
$$\sum_{t < N} \sum_{k \geq 1} |C_{tk}| = O(N). \tag{25}$$

However, given a problem (i.e., a transition matrix Q and an initial state distribution u), not every instance of TD(C_{tk}) can yield an anti-stable $\bar{A}$. In this section, we will show that for TD(λ), TD(λ, μ), and TD(C_k) (Section 3.3), no matter what Q and u are, $\bar{A}$ is always anti-stable, provided that X is non-singular and every non-absorbing state is reachable (i.e., $u(I - Q)^{-1} > 0$).

In this section, we will consider that $C_{tk} = C_k$ for all t. In such a case, Condition (25) is obviously satisfied if $\sum_k |C_k| < \infty$. Moreover, (24) reduces to

$$\bar{A} = X^T X\, Diag(u(I - Q)^{-1}) \sum_{k \geq 1} C_k (I - Q^k). \tag{26}$$

Let us formulate the general mathematical problem we will consider as follows: *What are the (desirably, necessary and sufficient) conditions on C_k such that for any u, any Q, and any positive definite matrix $P = X^T X$, the matrix $\bar{A}$ is anti-stable?* Sutton and Dayan have studied the case of TD(λ), and attempted to prove the convergence of TD(λ). However,

there is a flaw in their proofs. The flaw concerns a lemma used by both of them. The lemma, which was cited from [11] and given on Page 27 of [9], is as follows: *If S is a real, symmetric, and strictly diagonally dominant matrix with positive diagonal entries, then S is positive definite.* This lemma is correct, but their interpretation of "*strictly diagonally dominant*" was unfortunately wrong. Matrix S is strictly diagonally dominant if $|S_{i,i}| > \sum_{j \neq i} |S_{i,j}|$ for *all* i. They thought the above strict inequality was only needed for at least one i. A counter-example is that $S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}$. However, if the above strict inequality holds for at least one i and S is *irreducible*, then S is also positive definite [11]. But, the S matrix of concern in the proofs is not necessarily irreducible, for example, in the case where a non-absorbing state i is not reachable from any other states but $u_i > 0$.

5.1 Background in Linear Algebra

Definition 8 *Matrix A is robust anti-stable (RAS) if for any matrix $P > 0$,*

$$\sigma(PA) \subset z \in C : Re(z) > 0.$$

Matrix A is strongly robust anti-stable (SRAS) if for any matrix $P > 0$ and positive diagonal matrix D,

$$\sigma(DPA) \subset z \in C : Re(z) > 0.$$

Let us introduce two notations. $Ker(M)$ is the kernel of matrix M:

$$Ker(M) \triangleq \{x \mid Mx = 0\}.$$

$M^{\perp}$ is an orthogonal compliment of M.

Theorem 4 *Matrix A is RAS iff*

1. $R = A + A^* \geq 0$, *and*
2. $$A = \begin{matrix} \\ \end{matrix} \begin{matrix} Ker(R) & Ker(R^{\perp}) \end{matrix} \\ \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \qquad (27)$$

   where $A_{11} = 0$ and $Ker(A_{21}) = \{0\}$.

Proof: Necessary condition. Suppose A is RAS. First we prove that $R = A + A^* \geq 0$. Suppose that, in contrary, for some real non-zero vector x, $< (A + A^*)x, x > < 0$ and $< x, x > = 1$. Then, also $< Ax, x > < 0$. One can choose an orthogonal basis $\{e_0 = x, e_1, e_2, \cdots, e_n\} \triangleq U$. Here U is an unitary matrix with columns, $e_0, \cdots, e_n$. Then

$$A = U\tilde{A}U^* \text{ and } \tilde{A}(0,0) < 0.$$

If $P = UDiag(d_i)U^*$, then $P > 0$ if $d_i > 0$, and $tr(PA) = tr(Diag(d_i)\tilde{A}) = \sum_i d_i \tilde{A}(i,i)$. Since $\tilde{A}(1,1) < 0$, $tr(PA) < 0$. if $d_0 \gg d_1, \cdots, d_n$. We got a contradiction with a RAS property; that is, $tr(PA)$ should be positive. If $ker(A + A^*) = \{0\}$ (i.e., $A + A^* > 0$), then (27) obviously holds. Suppose that $Ker(A + A^*) = z \neq \{0\}$ and

$$A = \begin{array}{c} \begin{array}{cc} z & z^\perp \end{array} \\ \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \end{array}.$$

Since $z = Ker(A + A^*)$, $A_{11} + A_{11}^* = 0$, $A_{21} + A_{12}^* = 0$, and $A_{22} + A_{22}^* = 0$. Suppose that $A_{11} \neq 0$. Since A_{11} is skew symmetric, i.e., $A_{11} = -A_{11}^*$, there exists a unitary matrix U such that

$$A = \alpha\, U \left(\begin{array}{cc|c} 0 & 1 & \cdot \\ -1 & 0 & \\ \hline & C & \cdot \end{array}\right) U^*$$

where $\alpha > 0$.

Let us consider the following positive definite matrix

$$P = U \left(\begin{array}{c|c} \lambda I & TC^* \\ \hline -CT^* & I \end{array}\right) U^*$$

where $lambda \gg 1$ and $T = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$. Then

$$PA = \alpha\, U \left(\begin{array}{c|c} (\lambda I + CC^*)T & \cdot \\ \hline 0 & \cdot \end{array}\right) U^*$$

So $\alpha\, \sigma((\lambda I + CC^*)T) \subset \sigma(PA)$. But $((\lambda I + CC^*)T)_Q^* = (\lambda I + CC^*)T^*(\lambda I + CC^*)(\lambda I + CC^*)^{-1} = -(\lambda I + CC^*)T$, where $Q = (\lambda I + CC^*)^{-1} > 0$. So all eigenvalues of $(\lambda I + CC^*)T$ are pure imaginary ones, which contradicts to the assumption that A is RAS. So we have proved that $A_{11} = 0$. If $ker(A_{21}) \neq \{0\}$, then $Ker(A) \neq \{0\}$, which also contradicts to the RAS property.

Sufficient condition. We will use the following generalization of Lyapunov Theorem: If for some positive definite R, $RA + A^*R = Q \geq 0$ and the pair (A, Q) is *observable*, then all eigenvalues of A have a positive real part. Let us consider some positive definite P. Then

$$P^{-1}(PA) + (A^*P)P^{-1} = A + A^*,$$

$$Ker(A + A^*) = z$$

and

$$A = \begin{array}{c} \begin{array}{cc} z & z^\perp \end{array} \\ \begin{pmatrix} 0 & \cdot \\ A_{21} & \cdot \end{pmatrix} \end{array}.$$

where $Ker(A_{21}) = 0$. We have to prove that the pair $(PA, A + A^*)$ is observable. Indeed,

$$P = \begin{array}{c} \begin{array}{cc} z & z^\perp \end{array} \\ \left(\begin{array}{cc} D & Q \\ Q^T & B \end{array} \right) \begin{array}{c} z \\ z^\perp \end{array} \end{array},$$

$D > 0$, $B > 0$, and $D - QB^{-1}Q^T > 0$.

$$PA = \left(\begin{array}{c|c} D & Q \\ \hline Q^T & B \end{array} \right) \left(\begin{array}{c|c} 0 & \cdot \\ \hline A_{21} & \cdot \end{array} \right) = \left(\begin{array}{c|c} QA_{21} & \cdot \\ \hline BA_{21} & \cdot \end{array} \right).$$

Since $Ker(A_{21}) = \{0\}$ and $B > 0$, $Ker(BA_{21}) = \{0\}$. We just refer to a well-known result (see, for instance, [4]); that is, (A, H) is observable iff (A_{11}, A_{21}) is observable, where

$$P = \begin{array}{c} \begin{array}{cc} Ker(H) & Ker(H)^\perp \end{array} \\ \left(\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \end{array} \right) \begin{array}{c} Ker(H) \\ Ker(H)^\perp \end{array} \end{array}.$$

□

Example:[1] The matrix $A = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}$ is robust anti-stable (RAS). Indeed,

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}^T = \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} \triangleq L \geq 0,$$

$$Ker(L) = \{(\alpha, -\alpha)^T \mid \alpha \in R\},$$

and

$$Ker(L)^\perp = \{(\beta, \beta)^T \mid \beta \in R\}.$$

Then in the basis $e_1 = (1/\sqrt{2}, -1/\sqrt{2})^T$ and $e_2 = (1/\sqrt{2}, 1/\sqrt{2})^T$, matrix A has the following representation:

$$A = \begin{array}{c} \begin{array}{cc} e_1 & e_2 \end{array} \\ \left(\begin{array}{cc} 0 & \cdot \\ -1 & \cdot \end{array} \right) \begin{array}{c} e_1 \\ e_2 \end{array} \end{array},$$

$e_1 \in Ker(A + A^*)$ and $e_2 \in Ker(A + A^T)^\perp$. So A is RAS. We will see the same example later in connection with a different question.

Remark: If A is RAS, then obviously A is nonsingular. Moreover, A^{-1} is also RAS. Indeed for positive definite

$$\sigma(PA^{-1}) \subset \{z \in C : Re(z) > 0\} \ \ iff \ \ \sigma((PA^{-1})^{-1}) \subset \{z \in C : Re(z) > 0\}.$$

But $\sigma((PA^{-1})^{-1}) = \sigma(AP^{-1}) = \sigma(P^{-1}A)$, and $\sigma(P^{-1}A) \subset \{z \in C : Re(z) > 0\}$ since $P^{-1} > 0$. By an analogous argument, if A is RAS, then A^* is also RAS. But sum of the two RAS matrices is not necessary RAS. Indeed, consider a RAS matrix A such that $A + A^*$ is singular. A and A^* are RAS, but $A + A^*$ is not.

Proposition 3 *Consider an absorbing matrix Q (i.e., $Q_{i,j} \geq 0$, $\sum_j Q_{i,j} \leq 1$, and $\rho(Q) < 1$). For an arbitrary non-negative row $u = (u_1, u_2, \cdots, u_i, \cdots)$, we define $d = (d_1, d_2, \cdots, d_i, \cdots) = u(I-Q)^{-1}$, $d_i > 0$. Then $B \triangleq Diag^{-1}(d)Q^T Diag(d)$ is also semi-stochastic (i.e., $B_{i,j} \geq 0$ and $\sum_j B_{i,j} \leq 1$).*

Proof:

$$\sum_j B_{i,j} = \frac{1}{d_i} \sum_j Q_{j,i} d_j = \frac{d_i - u_i}{d_i} \leq 1$$

since $u_i \geq 0$. We used here that $d(I-Q) = u \ \ iff \ \ d_i - \sum_j Q_{j,i} d_j = u_i$ for all i. □

Corollary 4

$$\|Q\|_D \leq 1.$$

Proof: According to (19),

$$\|Q\|_D = \rho(QD^{-1}Q^T D).$$

Q and $D^{-1}Q^T D$ are semi-stochastic. So product $QD^{-1}Q^T D$ is also semi-stochastic. The latter implies that $\rho(QD^{-1}Q^T D) \leq 1$. □

Corollary 5

$$D(I-Q) + (I-Q^T)D \ > \ 0.$$

Proof: According to (18), it is equivalent to show that

$$(I-Q) + (I-Q^T)^*_D \ >_D \ 0.$$

Since $\|Q\|_D = \|Q^*_D\| \leq 1$, $(I-Q) + (I-Q)^*_D = 2I - (Q + Q^*_D) \ \geq_D \ 0$. Also we know that matrix $B = \frac{1}{2}(Q + Q^*_D)$ is semi-stochastic. Moreover, both Q and Q^*_D are semi-stochastic. Suppose that $I - \frac{1}{2}(Q + Q^*_D)$ is singular. Then by the famous Perron-Frobenius Theorem, for non-negative matrices, there exists nonzero non-negative vector $x = (x_1, x_2, \cdots, x_i, \cdots)$ such that $x[\frac{1}{2}(Q + Q^*_D)] = x$. Assume that $x_1, x_2, \cdots, x_\tau > 0$, and $x_{\tau+1}, \cdots, x_n = 0$. For Q (and similarly for Q^*_D), $\sum_{j=1} nQ(i,j) = 1$ when $1 \leq i \leq \tau$, and $Q(i,j) = 0$ when $1 \leq i \leq \tau$ and $j > \tau$. But this contradicts $\rho(Q) < 1$. □

Corollary 6

$$D(I-Q^n) + (I-(Q^n)^T)D \ > \ 0$$

where

$$D = Diag(d), \quad d > 0,$$

and

$$d = u(I-Q)^{-1}, \quad u \geq 0.$$

Proof: According to Corollary 5, it is enough to show that $d = \hat{u}(I-Q^n)^{-1}$ for some $\hat{u} \geq 0$. We use here that Q^n is also absorbing. Or it is equivalent to show that $d(I-Q^n) \geq 0$. But $d(I-Q^n) = u(I-Q)^{-1}(I-Q^n) = u(I+Q+Q^2+\cdots+Q^{n-1}) \geq 0$, since $u \geq 0$ and all powers Q^i have nonnegative elements. □

Corollary 7 *For any nonzero sequence of C_k such that $C_k \geq 0$ and $\sum C_k < \infty$,*

$$D(\sum_{k\geq 1} C_k(I - Q^k)) + (\sum_{k\geq 1} C_k(I - Q^k))D \ > \ 0.$$

Proof: It is a direct consequence of Corollary 6. □

Corollary 8

$$D(I + Q^k) + (I + Q^k)^T D \ > \ 0.$$

Proof: We prove a bit general results. If A is symmetric with non-negative elements and $(I - A) > 0$, then $(I + A) > 0$. Since A is symmetric, $\|A\| = \rho(A)$, and there exists a vector x with non-negative elements such that $Ax = \rho(A)x$. Then $0 < (I - A)x$, $x = (1 - \rho(A))\|x\|^2$, so $\rho(A) < 1$ and $-I < A < I$. Returning to our original problem, we use symmetricity and positive definiteness in terms of D-inner product. □

Proposition 4

$$A + A^* > 0 \ \ iff \ \ A^{-1} + (A^{-1})^* > 0.$$

Proof: First of all, if $A + A^* > 0$, then by Lyapunon Theorem all eigenvalues of A have positive real parts. So at least A^{-1} exists. Below we use the following well-known and very useful implication:

$$(P > 0) \wedge (det(X) \neq 0) \implies XPX^* > 0.$$

The following identities prove this proposition:

$$0 < A^{-1}(A + A^*)(A^*)^{-1} = (A^*)^{-1} + A^{-1} = A^{-1} + (A^{-1})^*.$$

□

Remark: It is clear that Proposition 4 works for any $P > 0$ using the following substitution:

$$A^* \Longrightarrow A_P^* \ \ and \ \ > \implies >_P .$$

Definition 9 *An invertible cone of functions is a set C of functions which satisfy the following two properties:*

1. $f_1, f_2 \in C \implies \alpha f_1 + \beta f_2 \in C$ *for any* $\alpha, \beta \geq 0$ *and* $\alpha^2 + \beta^2 > 0$.
2. $f \in C \implies \frac{1}{f} \in C$.

We use $C(f_1, \cdots, f_n)$ to denote the minimal invertible cone containing all functions f_i.

Proposition 5 *Suppose that the following inequality holds for some matrix Q and some functions f_i:*

$$f_i(Q) + (f_i(Q))^* > 0, \quad 1 \leq i \leq n,$$

then for any function $f \in C(f_1, \cdots, f_n)$

$$f(Q) + (f(Q))^* > 0.$$

Proof: This proposition is just a direct application of Proposition 4 and convexity of a set of positive definite matrices. □

Theorem 5 *Consider an arbitrary function f which belongs to invertible cone C_Q generated by 1 and $1 \pm x^n$, then*

$$Df(Q) + f(Q^T)D > 0$$

where Q is semi-stochastic, $\rho(Q) < 1$, $D = Diag(d_i)$, $d(I - Q) \geq 0$, and $d_i > 0$.

Proof: This theorem follows directly the previous corollaries. □

Proposition 6 (Caley Transform) *Suppose that $I + X$ is invertible, then*

$$(I - X)(I + X)^{-1} + ((I - X)(I + X)^{-1})^* > 0 \ \ iff \ \ \|X\| < 1, \tag{28}$$

and

$$(I - X)(I + X)^{-1} + ((I - X)(I + X)^{-1})^* \geq 0 \ \ iff \ \ \|X\| \leq 1. \tag{29}$$

Proof: We prove only (28). (29) can be proved similarly. Let us recall that for nonsingular matrix B, the following equivalence holds:

$$A > 0 \iff XAX^* > 0.$$

So

$$\begin{aligned}
& (I + X)^{-1}(I - X) + (I - X)^*((I + X)^{-1})^* > 0 \\
\iff & (I + X)((I + X)^{-1}(I - X) + (I - X)^*((I + X)^{-1})^*)(I + X)^* >=; 0 \\
\iff & (I - X)(I + X)^* + (I + X)(I - X)^* > 0 \\
\iff & I - X + X^T - XX^* + I + X - X^* - XX^* > 0 \\
\iff & XX^* < I \\
\iff & \|X\| < 1.
\end{aligned}$$

□

Corollary 9 *For any complex number x, $|z| = 1$,*

$$(zI - X)(zI + X)^{-1} + ((zI - X)(zI + X)^{-1})^* \geq 0 \ \ iff \ \ \|X\| \leq 1,$$

assuming $(zI + X)$ is invertible, which obviously holds if $\rho(X) < 1$. The above also holds when $\geq$ is replaced by $>$.

given a current state. In this section, we make the problem more general: The Markov chain can generate a non-zero reward at any state, and the learning task is, given a current state i, predicting the total rewards at the end, $V_i = X_i^T \cdot w$. Below we define a new weight-update operator for this type of learning tasks. With this operator, we will show that the mean of the prediction errors at epoch n is exactly the same as what we obtained before; that is, Equation (1). Because of the same error dynamics, the "convergence with probability one" property as well as other theorems apply to this type of learning tasks as well.

Let $\phi(s)$ be the "reward" received at state s. Since we have considered in the previous sections the case that $\phi(s) = 0$ for all non-absorbing states, here it is enough to consider only the case that

$$\phi(s) = 0 \ \ \forall s \in \mathcal{A}.$$

Let $\phi = (\phi(s_1), \phi(s_2), \cdots, \phi(s_n))^T$, where s_i is a non-absorbing state and n is the total number of non-absorbing states. As usual, let V^* be the perfect prediction.

Theorem 7

$$V^* = (I - Q)^{-1}\phi. \tag{30}$$

Proof:

$$\begin{aligned} V_i^* &= E(\sum_{t=0}^{\infty} \phi(s(t)) | s(0) = i \in \mathcal{N}) \\ &= I_i[I + Q + Q^2 + Q^3 + \cdots]\phi \\ &= I_i(I - Q)^{-1}\phi \end{aligned}$$

where I_i is a n-dimensional vector whose elements are all 0 except for the i-th element. Hence, the above theorem follows. □

Now we define an update operator to predict cumulative rewards:

Definition 10

$$\Delta w_{tk} \triangleq X_{s(t+k)}^T w - X_{s(t)}^T w + \phi(s(t)) + \phi(s(t+1)) + \cdots + \phi(s(t+k-1))X_{s(t)} \tag{31}$$

Theorem 8 *Corollary 1 still holds for the new operator in predicting cumulative rewards.*

Proof: In a similar way to prove Lemma 2, we now establish something similar to Lemma 2. (Recall that $z = 0$ now.)

$$\begin{aligned} & E(\Delta w_{tk} | \mu, T) \\ = \ & XDiag(uQ^t)(Q^k - I)X^T w + XDiag(uQ^t)[I + Q + Q^2 + \cdots + Q^{k-1}]\phi \\ = \ & XDiag(uQ^t)(I - Q^k)((I - Q)^{-1}\phi - X^T w) \\ = \ & XDiag(uQ^t)(I - Q^k)(V^* - X^T w) \end{aligned} \tag{32}$$

From (32), it is straightforward to prove the above theorem (see the proofs for Theorem 1 and Corollary 1). □

References

[1] A.G. Barto, S.J. Bradtke, and S.P. Singh. Real-time learning and control using asynchronous dynamic programming. Technical Report 91-57, Computer Science Department, University of Massachusetts, 1991.

[2] P. Dayan. The convergence of TD(λ) for general λ. *Machine Learning*, 8:341–362, 1992.

[3] P. Dayan and T.J. Sejnowski. TD(λ): Convergence with probability 1. *Machine Learning*, 14(3), 1994.

[4] L. Gurvits, T. Shalom, and L. Rodman. Controllability and completion of partial upper triangular matrices over rings. *Linear Algebra and Applications*, pages 135–149, 1992.

[5] A. hiryev. *Theory of Probabilities.* 1980.

[6] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8:293–321, 1992.

[7] Long-Ji Lin. Scaling up reinforcement learning for robot control. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 182–189. Morgan Kaufmann, 1993.

[8] A.L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal on Research and Development*, 3:210–229, 1959. Reprinted in E.A. Feigenbaum and J. Feldman (Eds.) *Computers and Thought*, 71-105, New York: McGraw-Hill, 1963.

[9] R.S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.

[10] G. Tesauro. Practical issues in temporal difference learning. *Machine Learning*, 8:257–277, 1992.

[11] R.S. Varga. *Matrix Iterative Analysis.* Englewood Cliffs, NJ: Prentice-Hall, 1962.

[12] C.J.C.H. Watkins. *Learning from Delayed Rewards.* PhD thesis, University of Cambridge, England, 1989.

[13] M.H. Wonham. *Linear multivariable Control: A Geometric Approach.* Springer Verlog, Berlin, 1979.