Stochastic Shortest Path Games and Q-Learning∗

Huizhen Yu†

Abstract

We consider a class of two-player zero-sum stochastic games with finite state and compact control spaces, which we call stochastic shortest path (SSP) games. They are undiscounted total cost stochastic dynamic games that have a cost-free termination state. Exploiting the close connection of these games to single-player SSP problems, we introduce novel model conditions under which we show that the SSP games have strong optimality properties, including the existence of a unique solution to the dynamic programming equation, the existence of optimal stationary policies, and the convergence of value and policy iteration. We then focus on finite state and control SSP games and the classical Q-learning algorithm for computing the value function. Q-learning is a model-free, asynchronous stochastic iterative algorithm. By the theory of stochastic approximation involving monotone nonexpansive mappings, it is known to converge when its associated dynamic programming equation has a unique solution and its iterates are bounded with probability one. For the SSP case, as the main result of this paper, we prove the boundedness of the Q-learning iterates under our proposed model conditions, thereby establishing completely the convergence of Q-learning for a broad class of total cost finite-space stochastic games.

∗This work was supported by the Air Force Grant FA9550-10-1-0412.
†Huizhen Yu was with the Laboratory for Information and Decision Systems (LIDS), MIT, and she is now with the Department of Computing Science, University of Alberta. janey.hzyu@gmail.edu
Contents

1 Introduction 3

2 A Finite-State Compact-Control SSP Game Model 4
   2.1 Basic Definitions and Conditions 4
   2.2 An SSP Game Model and its Optimality Properties 7
   2.3 Further Remarks 11

3 Q-Learning for Finite-Space SSP Games 12
   3.1 Finite-Space SSP Games 13
   3.2 Q-Learning for SSP Games 16

4 Boundedness of Q-Learning Iterates 20
   4.1 Boundedness Analysis for the General Case 20
      4.1.1 Relate \( \{Q_t\} \) to Q-learning type iterations in a single-player SSP problem 21
      4.1.2 Auxiliary sequence \( \{\tilde{Q}_t\} \) . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
      4.1.3 Some notation and definitions 24
      4.1.4 Choose \( t_0 \) and initial \( \tilde{Q}_{t_0} \) for a sample path and \( \delta > 0 \) . . . . . . 25
      4.1.5 Interpret \( \{\tilde{Q}_t\} \) as total costs in certain SSP problems neighboring SSP(\( \bar{\nu} \)) . . 27
      4.1.6 Lower boundedness of \( \{\tilde{Q}_t\} \) . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
   4.2 Boundedness Analysis for a Special Case . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

References 31

Appendix A Proof of Lemma 4.35 33
1 Introduction

In this paper we consider two-player zero-sum stochastic dynamic games under the undiscounted, total cost criterion, and we focus on those games that have a finite state space and a cost-free termination state. Our interest is in using a well-known model-free stochastic approximation algorithm, the Q-learning algorithm, for computing the value of a game when the control spaces of both players are finite. The main purpose of this paper is to show that there is a broad class of total cost games with desirable optimality properties for which the Q-learning algorithm converges in a totally asynchronous setting under fairly mild conditions.

Zero-sum stochastic games were first introduced by Shapley [Sha 53] for the discounted cost criteria. Since then there have been extensive research on undiscounted stochastic games, including games with the limiting average cost criterion, first considered by Gillette [Gil 57] and developed in the seminal works [BF68, BK76, MNS1], and games with total cost and related criteria [TV87, Feli80, Now85, Now99]. (We refer readers to the excellent book by Filar and Vrieze [FV97] for historical and contemporary developments on stochastic games.) A general formulation of total cost games, when one-stage costs can be positive or negative-valued, was first proposed and analyzed by Thuijsman and Vrieze [TV87]. In this and their subsequent works (see the survey by Thuijsman and Vrieze [TV98] and also Filar and Vrieze [FV97, Chapter 4]), they established important existence results for finite state and control total cost games. They showed that for a total cost game to have a finite value function, a sufficient condition is that the corresponding average-cost game has the value zero and both players possess stationary average-cost optimal policies. They also showed that for a total cost game to have not only a finite value function but also stationary optimal policies for both players, a necessary and sufficient condition is that a certain system of functional equations have solutions.

In this paper we will focus on a subset of the total cost games of the latter kind. In addition to having a value and stationary optimal policies, the SSP games we consider also have the property that their associated Bellman equation has a unique solution. This property relates to the convergence of value iteration and is essential for the Q-learning algorithm we are interested in. Among the total cost games satisfying Thuijsman and Vrieze’s necessary and sufficient conditions mentioned earlier, the ones that will be excluded from our consideration are, briefly speaking, those games in which from some initial state, both players can play some stationary optimal policies (and incur zero average cost) without ever reaching the termination state. (We will discuss in Section 2.3 some examples of such games.)

To delineate a subset of SSP games with desirable properties, we will specify conditions on the model of the games, and we will do so in the broader context of games with compact control sets and semi-continuous one-stage costs, which include finite-control games as special cases. (A finite-control game can be viewed as a game with compact control sets, where controls correspond to randomized decision rules of each player.) In the context of total cost compact-control games, there are several earlier works [KC69, KS81, PB99], and the one by Patek and Bertsekas [PB99] is most related to ours. They considered finite-state compact-control SSP games in which one-stage costs can take both positive and negative values, and the termination state need not be reachable for every initial state and every pair of policies of the two players. The term “SSP games” is, in fact, from [PB99], and it is based on the close connection of SSP games, at both analytical and computational level, to single-player SSP problems, which are total cost or total reward Markov decision processes (MDP) with a termination state. (For references on SSP and total cost MDP, see e.g., Bertsekas and Tsitsiklis [BT91, BT96], Feinberg [Fei92], and Puterman [Put94].) Patek and Bertsekas [PB99] established optimality results similar to those we aim to obtain, but under model conditions that are asymmetric in terms of the two players and bear a strong association with pursuit-evasion type of games.

As one of the contributions of this paper, we introduce a symmetric formulation of model con-
ditions (Assumption 2.3). It characterizes a much broader class of SSP games than considered in [PB99] (see Section 2.3 for a detailed comparison), and it allows the theory of single-player SSP problems (Bertsekas and Tsitsiklis [BT91]) to be more fully utilized in analyzing the compact-control SSP games. As a result, we show that the desired optimality properties, including the existence of a unique solution to the Bellman equation and the existence of a pair of equilibrium policies that are stationary deterministic (Theorem 2.1), as well as the convergence properties of value iteration and policy iteration (Theorem 2.2), are retained.

We then consider finite state and control SSP games that satisfy the proposed model conditions, and we turn to the question of the convergence of the Q-learning algorithm for computing their value functions. Q-learning was first introduced by Watkins [Wat89] in the context of MDP and reinforcement learning, and its convergence was analyzed most comprehensively by Tsitsiklis [Tsi94] as a special case of the convergence of asynchronous stochastic approximation algorithms. For discounted stochastic games, Littman [Lit96] studied Q-learning and analyzed its convergence (with a different argument than [Tsi94, Theorem 3], which also implies the convergence of Q-learning in such games). For undiscounted SSP games whose Bellman equation admits a unique solution, convergence of Q-learning is known in two limited cases under strong assumptions:

(i) when the game always terminates regardless how the two players play, and

(ii) when the iterates generated by Q-learning are bounded with probability one.

In both cases, the convergence of Q-learning follows from the convergence theorems of Tsitsiklis [Tsi94] for asynchronous stochastic approximation involving sup-norm contraction or monotone nonexpansive mappings: convergence in the first case is due to a contraction property (Patek and Bertsekas [PB99, Lemma 4.1]), and convergence in the second case (under the boundedness condition) follows from arguments for monotone nonexpansive mappings [Tsi94, Theorem 2]. (For more details, see Bertsekas and Tsitsiklis [BT96, Chapter 4 and Section 7.2].) Another convergence result is also known when boundedness of Q-learning iterates is not assumed, based on the results of Abounadi, Bertsekas and Borkar [ABB02]. However, in this case, additional conditions are required on the timing and frequency of component updates in Q-learning, which are more restrictive than the totally asynchronous computing framework of [Tsi94].

The main contribution of this paper is a boundedness proof for the Q-learning algorithm with totally asynchronous computation, for the broad class of SSP games satisfying our model conditions. We show that the Q-learning iterates are bounded with probability one (Theorem 3.1), thereby furnishing the boundedness condition required in the convergence theorem of [Tsi94] and establishing completely the convergence of Q-learning (Theorem 3.2). Our proof techniques are based on those constructed in Yu and Bertsekas [YB13a] for analyzing boundedness of Q-learning in single-player SSP problems.

This paper is organized as follows. In Section 2 we consider finite-state compact-control SSP games, and introduce our new model conditions and prove optimality results. In Section 3 we describe finite state and control SSP games and the Q-learning algorithm. Finally, in Section 4 we present the boundedness analysis for Q-learning.

2 A Finite-State Compact-Control SSP Game Model

2.1 Basic Definitions and Conditions

We consider a finite-state two-player zero-sum total cost stochastic game with a termination state. Let \( S_o = S \cup \{0\} \) be the state space, where \( S = \{1, \ldots, n\} \) and state 0 is a cost-free termination (absorbing) state. Two players participate in the game with opposite objectives, and their actions jointly influence the evolution of the states through time. In particular, at each state \( i \in S \), player I
(player II, respectively) can apply a control from a set $U(i)$ ($V(i)$, respectively) of feasible controls, where $U(i)$ and $V(i)$ are assumed to be compact sets in some complete separable metric space. If the two players apply a pair of controls $(\tilde{u}, \tilde{v}) \in U(i) \times V(i)$, an expected one-stage cost $c_i(\tilde{u}, \tilde{v})$ is incurred to player I while player II receives the same amount as an expected one-stage reward, and the system then transitions from state $i$ to state $j \in S_o$ with probability $p_{ij}(\tilde{u}, \tilde{v})$. Here the one-stage costs (with respect to player I) can be positive or negative. We assume that the transition probabilities and one-stage costs satisfy the following continuity/semi-continuity conditions:

**Assumption 2.1 (Continuity Condition).** For all states $i, j \in S$, the transition probability $p_{ij}(\tilde{u}, \tilde{v})$ is a continuous function on $U(i) \times V(i)$, and the one-stage cost $c_i(\tilde{u}, \tilde{v})$ is lower semicontinuous in $\tilde{u}$ for fixed $\tilde{v}$ and upper semicontinuous in $\tilde{v}$ for fixed $\tilde{u}$.

Starting from some state $i_0 \in S$ at time 0, the players play for an infinite number of stages, making control decisions based on the information of the current state and the history of the game, which includes the past states and past controls applied by each player, while the states evolve in a Markovian way as described above. We define the total costs for player I and the total rewards for player II as follows.

Let $i_k$ denote the state and $(\tilde{u}_k, \tilde{v}_k)$ the controls taken by the two players at time $k$. Let $\Pi_1, \Pi_2$ denote the sets of all history-dependent randomized policies for player I and player II, respectively (each of such policies is a collection of Borel measurable transition probabilities from the space of histories to the respective player’s control space). If player I adopts policy $\pi_1 \in \Pi_1$ and player II $\pi_2 \in \Pi_2$, we define the total cost of player I (total reward of player II) for the initial state $i_0 = i$ by

$$x_i(\pi_1, \pi_2) = \liminf_{t \to \infty} E_{\pi_1, \pi_2} \left[ \sum_{k=0}^{t} c_k(\tilde{u}_k, \tilde{v}_k) \mid i_0 = i \right],$$

where $\{(i_k, \tilde{u}_k, \tilde{v}_k), k \geq 0\}$ is the random process of states and controls induced by the policy pair $(\pi_1, \pi_2)$, and $E_{\pi_1, \pi_2}$ denotes expectation with respect to the probability distribution of the induced process. In vector notation we write $x(\pi_1, \pi_2)$ for the vector of total costs, $(x_1(\pi_1, \pi_2), \ldots, x_n(\pi_1, \pi_2))$.

The optimal total cost for player I and optimal total reward for player II, for each initial state $i \in S$, are defined to be

$$\bar{x}_i^* = \inf_{\pi_1 \in \Pi_1} \sup_{\pi_2 \in \Pi_2} x_i(\pi_1, \pi_2), \quad \underline{x}_i^* = \sup_{\pi_2 \in \Pi_2} \inf_{\pi_1 \in \Pi_1} x_i(\pi_1, \pi_2),$$

respectively. An optimal policy for player I (player II) is then a policy which attains the optima for all states in the above minimization over $\Pi_1$ (maximization over $\Pi_2$). We call $\bar{x}_i^*, \underline{x}_i^*$ the upper and lower value of the game for state $i$. If these values coincide for all states, we call the corresponding $x^* = (x_1^*, \ldots, x_n^*)$ where $x_k^* = \bar{x}_k^* = \underline{x}_k^*$, the value function of the game. We say that $(\pi_1^*, \pi_2^*) \in \Pi_1 \times \Pi_2$ is a pair of equilibrium policies if the following holds:

$$x(\pi_1^*, \pi_2^*) \leq x(\pi_1^*, \pi_2) \leq x(\pi_1, \pi_2^*), \quad \forall \pi_1 \in \Pi_1, \pi_2 \in \Pi_2.$$

In that case $x^* = x(\pi_1^*, \pi_2^*)$ is the value function of the game, and $\pi_1^*, \pi_2^*$ are optimal policies for the two players.

Consider the class of stationary deterministic policies of each player, which is defined for player I and player II by

$$D_1 = \left\{ \mu : S \mapsto \bigcup_{i \in S} U(i) \mid \mu(i) \in U(i), \quad i \in S \right\},$$

$$D_2 = \left\{ \nu : S \mapsto \bigcup_{i \in S} V(i) \mid \nu(i) \in V(i), \quad i \in S \right\},$$

respectively. Each function $\mu \in D_1$ corresponds to a policy that applies at time $k$ the control $\mu(i_k)$ for state $i_k$, and this policy will also be denoted by $\mu$. We use similar notation for the policies
corresponding to $D_2$. We will shortly introduce model conditions that guarantee the existence of equilibrium policies within these policies.\footnote{Without loss of generality, we focus on stationary deterministic policies here instead of stationary randomized policies, because our results can be applied in compact-control problems after a reformulation that let $\bar{U}(i)$ and $\bar{V}(i)$ represent probability distributions over the actual control sets.}

With stationary policies in $D_1, D_2$, we define several dynamic programming operators on $\mathbb{R}^n$ for the game, using compact matrix and vector notation. For a pair of policies $(\mu, \nu) \in D_1 \times D_2$, let $T_{\mu \nu} : \mathbb{R}^n \to \mathbb{R}^n$ be given by

$$T_{\mu \nu} x = c(\mu, \nu) + P(\mu, \nu)x, \quad x \in \mathbb{R}^n,$$

where $c(\mu, \nu)$ is the $n$-dimensional one-stage cost vector with components $c_i(\mu(i), \nu(i))$, and $P(\mu, \nu)$ is the $n$-by-$n$ substochastic transition probability matrix with elements $[P(\mu, \nu)]_{ij} = p_{ij}(\mu(i), \nu(i))$, $i, j \in S$. Define $T_\mu : \mathbb{R}^n \to \mathbb{R}^n$ and $T_\nu : \mathbb{R}^n \to \mathbb{R}^n$ by

$$T_\mu x = \sup_{\nu \in D_2} \{ c(\mu, \nu) + P(\mu, \nu)x \}, \quad T_\nu x = \inf_{\mu \in D_1} \{ c(\mu, \nu) + P(\mu, \nu)x \}.$$

In the right-hand sides above the optimization over $D_1$ or $D_2$ is component-wise.\footnote{Here we use the matrix/vector notation to write $n$ optimization problems in one expression. This is valid because of the separable structure of these problems. For example, the problem of maximizing the $i$th component of $c(\mu, \nu) + P(\mu, \nu)x$ over $D_2$ is identical to $\sup_{\nu \in \bar{V}(i)} [c_i(\mu(i), \nu(i))] + \sum_{j \in S} p_{ij}(\mu(i), \nu(i))x_j$. In other words, the $i$th optimization problem depends only on the components of $\mu, \nu$ for state $i$.}

Finally, we define $T : \mathbb{R}^n \to \mathbb{R}^n$ and $\bar{T} : \mathbb{R}^n \to \mathbb{R}^n$ by

$$T x = \inf_{\mu \in D_1} \sup_{\nu \in D_2} \{ c(\mu, \nu) + P(\mu, \nu)x \}, \quad \bar{T} x = \sup_{\nu \in D_2} \inf_{\mu \in D_1} \{ c(\mu, \nu) + P(\mu, \nu)x \},$$

where, similar to the above, the optimization in the right-hand side is component-wise.

A mapping $H$ is monotone if $Hx \leq Hy$ for $x \leq y$. Since $P(\mu, \nu)$ is a nonnegative matrix, the above mappings are monotone by definition. They also satisfy, be definition,\footnote{Assumption \ref{assumption:continuity} on the continuity of the state transition probabilities and the semicontinuity of the one-stage costs implies that every component of $T_\mu x$ is lower semicontinuous in $(x, \mu)$, every component of $\bar{T}_\mu x$ is upper semicontinuous in $(x, \nu)$, and every component of $T_{\mu \nu} x$ is lower semicontinuous in $(x, \mu)$ for fixed $\nu$ and upper semicontinuous in $(x, \nu)$ for fixed $\mu$. Since the control sets are compact, it then follows that under Assumption \ref{assumption:continuity} the infimum and supremum in the definitions of the above mappings are all attained: for every $x$, there exists $\mu$ such that $Tx = T_\mu x$; for every $x$ and $\mu$, there exists $\nu$ such that $T_\mu x = T_{\mu \nu} x$; and similar relations hold for $\bar{T}$ and $\bar{T}_\mu$.}

\begin{align*}
T_\mu x \leq T_{\mu \nu} x \leq T_\nu x, & \quad \forall \mu \in D_1, \nu \in D_2, x \in \mathbb{R}^n, \\
\bar{T}_\mu x \leq \bar{T} x \leq \bar{T}_\nu x, & \quad \forall \mu \in D_1, \nu \in D_2, x \in \mathbb{R}^n.
\end{align*}

Furthermore, Assumption \ref{assumption:continuity} on the continuity of the state transition probabilities and the semicontinuity of the one-stage costs implies that every component of $T_\mu x$ is lower semicontinuous in $(x, \mu)$, every component of $\bar{T}_\mu x$ is upper semicontinuous in $(x, \nu)$, and every component of $T_{\mu \nu} x$ is lower semicontinuous in $(x, \mu)$ for fixed $\nu$ and upper semicontinuous in $(x, \nu)$ for fixed $\mu$. Since the control sets are compact, it then follows that under Assumption \ref{assumption:continuity} the infimum and supremum in the definitions of the above mappings are all attained: for every $x$, there exists $\mu$ such that $Tx = T_\mu x$; for every $x$ and $\mu$, there exists $\nu$ such that $T_\mu x = T_{\mu \nu} x$; and similar relations hold for $\bar{T}$ and $\bar{T}_\mu$.

We also need a regularity condition:

**Assumption 2.2** (Minimax Regularity Condition). For all $x \in \mathbb{R}^n$, we have $Tx = \bar{T} x$, i.e.,

$$\inf_{\mu \in D_1} \sup_{\nu \in D_2} T_{\mu \nu} x = \sup_{\nu \in D_2} \inf_{\mu \in D_1} T_{\mu \nu} x.$$
2.2 An SSP Game Model and its Optimality Properties

We now introduce a novel formulation of an SSP game model. We will show that it has favorable optimality properties, including the existence of a unique solution to the dynamic programming equation, the existence of a pair of stationary equilibrium policies, and convergence of value and policy iteration.

We will put model assumptions on the cost/reward of certain policies depending on whether the termination state can be reached with probability 1 (w.p.1, for short). We need the following definition, which uses terminologies from [PB99].

Definition 2.1 (Prolonging and Non-prolonging Policies). We say a pair of policies \((\pi_1, \pi_2)\) is prolonging, if under these policies of the two players, there is a positive probability that the termination state 0 is never reached for some initial state. Then, a non-prolonging pair \((\pi_1, \pi_2)\) is one such that under these policies, the termination state is reached for any initial state w.p.1.

Assumption 2.3 (SSP Game Model).

(i) There exists a policy \(\bar{\mu} \in D_1\) for player I such that for any policy \(\nu \in D_2\), \(x_i(\bar{\mu}, \nu) < +\infty\) for all states \(i\).

(ii) There exists a policy \(\bar{\nu} \in D_2\) for player II such that for any policy \(\mu \in D_1\), \(x_i(\mu, \bar{\nu}) > -\infty\) for all states \(i\).

(iii) For any pair of policies \((\mu, \nu) \in D_1 \times D_2\) that is prolonging, \(x_i(\mu, \nu) = +\infty\) or \(-\infty\) for at least one initial state \(i\).

Assumption 2.3 has a symmetric form for the two players. It is much broader than the asymmetric SSP model formulation in the earlier work [PB99], as we will explain in Section 2.3. Assumption 2.3(i)-(ii) says that each player has at least one stationary policy to safeguard against infinite loss. Assumption 2.3(iii) says that a prolonging policy pair \((\mu, \nu)\) will be against the interest of some player. It also implies that the pair \((\bar{\mu}, \bar{\nu})\) of policies described in Assumption 2.3(i)-(ii) cannot be prolonging.

To derive further implications of Assumption 2.3, we consider the decision problem for one player when the other player plays a fixed stationary policy. In that case, the problem of optimizing the total cost or reward for one player is a total cost or reward MDP with a cost-free termination state. For these finite-state compact-control MDP, strong optimality properties are known under certain assumptions on the total cost/reward structure (Bertsekas and Tsitsiklis [BT91]). Like [PB99] on SSP games, our SSP game model in Assumption 2.3 is also motivated by these analytical results for single-player problems.

More specifically, let us consider a single-player problem which, when viewed as a two-player game by assuming there is a second dummy player who has singleton control sets, satisfies the model description in Section 2.1 including the continuity conditions in Assumption 2.1. We will refer to such a problem as a single-player SSP problem, whether it is to minimize total costs or to maximize total rewards. For a single-player SSP, we have from [BT91] the following notion of proper policies and a model condition that uses this notion and leads to desirable optimality properties.

Definition 2.2 (Proper and Improper Policies in Single-Player SSP). In a single-player SSP problem, a policy is said to be proper if under that policy, the termination state is reached w.p.1 for any initial state; the policy is said to be improper, otherwise.

3 Because of the use of liminf, the definition of the total cost function \(x(\pi_1, \pi_2)\) for a pair of general policies \((\pi_1, \pi_2)\) is asymmetric for the two players. However, for a pair of stationary policies \((\mu, \nu) \in D_1 \times D_2\), it can be shown that under Assumption 2.3 the limit of the finite-stage costs (or rewards) always exists (it may be finite, \(+\infty\) or \(-\infty\)). Because of this, the model assumption we introduce is indeed fully symmetric in terms of the two players.
The results of [BT91] show that if a single-player SSP problem satisfies the following assumption,

**SSP Model Assumption:** In the class of stationary deterministic policies, there exists a proper policy, and every improper policy incurs cost $+\infty$ for at least one initial state.

then the optimal total cost function is finite, and it is the unique solution of the dynamic programming equation. Moreover, value iteration converges starting from any initial value.

Based on these results for single-player SSP, let us introduce a notion of well-behaved policies for each player in SSP games. Let us call a policy of player I or player II essentially proper if, when the player plays that policy, the resulting (total cost or total reward) single-player SSP problem for the other player satisfies the SSP Model Assumption. In other words:

**Definition 2.3 (Essentially Proper Policies),**

(a) $\mu \in D_1$ is essentially proper if there exists a policy $\nu \in D_2$ such that $(\mu, \nu)$ is non-prolonging, and moreover, for every policy $\nu \in D_2$ with $(\mu, \nu)$ being prolonging, $x_i(\mu, \nu) = -\infty$ for at least one initial state $i$;

(b) $\nu \in D_2$ is essentially proper if there exists a policy $\mu \in D_1$ such that $(\mu, \nu)$ is non-prolonging, and moreover, for every policy $\mu \in D_1$ with $(\mu, \nu)$ being prolonging, $x_i(\mu, \nu) = +\infty$ for at least one initial state $i$.

If player I plays an essentially proper policy $\mu$, the reward-maximization problem player II faces is a single-player total-reward SSP with its dynamic programming operator given by $T_\mu$ [cf. Eq. (2.2)]. Similarly, if player II plays an essentially proper policy $\nu$, then player I has a single-player total-cost SSP problem with its dynamic programming operator given by $T_\nu$ [cf. Eq. (2.2)]. Hence by [BT91] the essentially proper policies we just defined have the following property.

**Lemma 2.1.** Let $H = T_\mu$ or $T_\nu$, where $\mu \in D_1$ or $\nu \in D_2$ is essentially proper. Then the equation $x = Hx$ has a unique solution $\bar{x}$, and $\lim_{t \to \infty} H^t x = \bar{x}$ for all $x \in \mathbb{R}^n$.

For any pair of essentially proper policies of the two players, we have the following fact:

**Lemma 2.2.** Let $\mu \in D_1$ and $\nu \in D_2$ be essentially proper. Then,

(i) $(\mu, \nu)$ is non-prolonging; and

(ii) $\bar{x}(\mu) \geq \bar{x}(\nu)$, where $\bar{x}(\mu), \bar{x}(\nu)$ are the unique solution of $x = T_\mu x$ and $x = T_\nu x$, respectively.

**Proof.** To prove (i), first we note that although for a policy $\mu$ to be essentially proper, Definition 2.3(a) does not exclude that $x_i(\mu, \nu) = +\infty$ for some state $i$ and a prolonging policy pair $(\mu, \nu)$, this cannot happen. Otherwise, we can derive a contradiction by constructing a policy $\nu' \in D_2$ for player II such that $(\mu, \nu')$ is prolonging but $x_i(\mu, \nu') > -\infty$ for all states $i$. This policy $\nu'$ can be chosen as follows. Suppose $x_i(\mu, \nu) = +\infty$ for some state $i$. Then, for the Markov chain induced by $(\mu, \nu)$, by [Put94] Theorem 9.4.1, p. 472, there exists a recurrent class $E$ such that the average cost on $E$ is strictly greater than 0. Let $\nu'$ be identical to $\nu$ for states in $E$ and identical to a policy $\bar{\nu}$ for the rest of the states, where $\bar{\nu}$ is such that $(\mu, \bar{\nu})$ is non-prolonging and the existence of $\bar{\nu}$ is ensured by the definition of $\mu$ as an essentially proper policy. The Markov chain induced by $(\mu, \nu')$ has two recurrent classes, $E$ and $\{0\}$, so $(\mu, \nu')$ is prolonging, and moreover, the average cost on $E$, [which is equal to the average cost on $E$ under $(\mu, \nu)$], is strictly greater than 0. Then, by [Put94] Theorem 9.4.1, p. 472, $x_i(\mu, \nu') = +\infty$ for all $i \in E$, while for the rest of the states $i$, we have that either $x_i(\mu, \nu') = +\infty$ or $x_i(\mu, \nu')$ is finite. Hence, $x_i(\mu, \nu') > -\infty$ for all $i$, and since the pair of policies $(\mu, \nu')$ is prolonging, this contradicts the definition of $\mu$ being essentially proper. Similarly, in Definition 2.3(b), it cannot happen that $x_i(\mu, \nu) = -\infty$ for some state $i$ and a prolonging policy pair $(\mu, \nu)$ when $\nu$ is essentially proper.
On the other hand, when \( \mu \) and \( \nu \) are essentially proper, Definition \( 2.3 \) dictates that if \((\mu, \nu)\) were prolonging, there must exist some states \( i, j \) with \( x_i(\mu, \nu) = -\infty \) and \( x_j(\mu, \nu) = +\infty \), which is impossible as we just argued. Therefore, the pair \((\mu, \nu)\) must be non-prolonging.

We now prove (ii). Since \( \mu \) and \( \nu \) are essentially proper, by Lemma \( 2.1 \) the equations \( x = T_\mu x \) and \( x = T_\nu x \) have a unique solution. Denote \( \bar{x} = \bar{x}(\mu), \tilde{x} = \tilde{x}(\nu) \). Since \( \bar{x} = T_\mu \bar{x} \geq T_\nu \tilde{x} \) [cf. Eq. (2.4)] and \( T_\nu \) is monotone, we have that for all \( t, \bar{x} \geq T_\nu^t \bar{x} \). By Lemma \( 2.1 \), \( \lim_{t \to \infty} T_\nu^t \bar{x} = \tilde{x} \). Therefore \( \bar{x} \geq \tilde{x} \).

In terms of essentially proper policies, Assumption \( 2.3 \) has an important implication given below.

**Lemma 2.3.** Suppose Assumption \( 2.3 \) holds. Let \( \mu \in D_1 \) and \( \nu \in D_2 \). Then we have:

(i) If there exists \( x \in \mathbb{R}^n \) such that \( x \geq T_\mu x \), then \( \mu \) is essentially proper.

(ii) If there exists \( x \in \mathbb{R}^n \) such that \( x \leq T_\nu x \), then \( \nu \) is essentially proper.

Moreover, every player has at least one essentially proper stationary policy—\( \bar{\mu} \) for player I and \( \bar{\nu} \) for player II where \( \bar{\mu} \) and \( \bar{\nu} \) are as in Assumption \( 2.3 \)(i)-(ii).

**Proof.** We prove (i); the proof for (ii) is entirely symmetric. For any policy \( \nu \in D_2 \), since \( x \geq T_\mu x \geq T_\nu x [\text{cf. Eq. (2.4)}] \), by the monotonicity of \( T_\nu \) and the definition of \( x_i(\mu, \nu) \), we have that \( x_i(\mu, \nu) < +\infty \) for any state \( i \). Then for the policy \( \bar{\nu} \) of player II in Assumption \( 2.3 \)(ii), the pair \((\bar{\mu}, \bar{\nu})\) must be non-prolonging by Assumption \( 2.3 \)(iii), and also by Assumption \( 2.3 \)(iii), for every \( \nu \in D_2 \) such that \((\mu, \nu)\) is prolonging, \( x_i(\mu, \nu) = -\infty \) for at least one state \( i \). By Definition \( 2.3 \)(a), this shows that \( \mu \) is essentially proper.

We now prove the last statement of the lemma. Consider the policies \( \bar{\mu} \) and \( \bar{\nu} \) in Assumption \( 2.3 \)(i) and (ii). As discussed immediately after that assumption, the pair \((\bar{\mu}, \bar{\nu})\) is non-prolonging. Moreover, Assumption \( 2.3 \)(i) and (iii) together imply that for any policy \( \nu \in D_2 \) such that \((\bar{\mu}, \nu)\) is prolonging, we must have \( x_i(\bar{\mu}, \nu) = -\infty \) for some initial state \( i \). Hence \( \bar{\mu} \) is essentially proper for player I by Definition \( 2.3 \)(a). Similarly, the policy \( \bar{\nu} \) is essentially proper for player II by Assumption \( 2.3 \)(ii)-(iii) and Definition \( 2.3 \)(b).

We are now ready to establish the optimality properties for the proposed SSP game model. Some of the proof steps below appear similar to those in [PBP99].

**Theorem 2.1** (Existence of Value and Equilibrium Policies). Under Assumptions \( 2.1 \) and \( 2.3 \), the game has a finite value function \( x^* \), which is the unique solution of the dynamic programming equation \( x = Tx \). Furthermore, any \( \mu^* \in D_1, \nu^* \in D_2 \) such that \( x^* = T_{\mu^*} x^* = T_{\nu^*} x^* \) are essentially proper (hence \((\mu^*, \nu^*)\) is non-prolonging). Such policies exist, and they form a pair of equilibrium policies for the game and are optimal for each player.

**Proof.** We show first that \( T \) can have at most one fixed point. Suppose that both \( x \) and \( x' \) satisfy \( x = Tx \) and \( x' = Tx' \). Under Assumption \( 2.1 \) there exist stationary deterministic policies \( \mu \) and \( \mu' \) such that \( T_{\mu} x = Tx \) and \( T_{\mu'} x' = Tx' \). By Lemma \( 2.3 \) both \( \mu \) and \( \mu' \) are essentially proper. Since \( x = Tx \leq T_{\mu} x \) [cf. Eq. (2.3)], by the monotonicity of \( T_{\mu'} \), we have that for all \( t, x \leq T_{\mu'}^t x \). On the other hand, since \( x' = T_{\mu'} x' \) and \( \mu' \) is essentially proper, we have by Lemma \( 2.1 \) that \( \{T_{\mu}^t x \} \) converges to \( x' \). Therefore \( x \leq x' \). A symmetric argument yields \( x' \leq x \), and hence \( x = x' \).

We now show that \( T \) has a fixed point. Let \( \bar{\mu} \in D_1 \) and \( \bar{\nu} \in D_2 \) be essentially proper policies, which exist under Assumption \( 2.3 \) (Lemma \( 2.3 \)). By Lemma \( 2.1 \) there exist a unique \( \bar{x} \) such that \( \bar{x} = T_{\bar{\mu}} \bar{x} \) and a unique \( \bar{x} \) such that \( \bar{x} = T_{\bar{\nu}} \bar{x} \). By Lemma \( 2.3 \)(ii), \( \bar{x} \geq \bar{x} \). Since \( T_{\bar{\mu}} \bar{x} \geq T \bar{x} \) and \( T_{\bar{\nu}} \bar{x} \leq T \bar{x} \) [cf. Eq. (2.3)], we also have \( \bar{x} \geq T \bar{x} \) and \( \bar{x} \leq T \bar{x} \). Using the monotonicity of \( T \) and the fact that \( \bar{x} \geq \bar{x} \), it follows that \( \{T^t \bar{x} \} \) is a non-increasing sequence bounded below by \( \bar{x} \) and hence
converges to some \( x^* \). Since \( T^{t+1} \bar{x} = T(T^t \bar{x}) \), by the continuity of \( T \), \( x^* \) must satisfy \( x^* = Tx^* \). Thus \( x^* \) is a fixed point of \( T \) and hence the unique fixed point of \( T \).

Now let \((\mu^*, \nu^*)\) be stationary deterministic policies such that \( T\mu^* \pi = T\nu^* \pi \) and \( T\nu^* \pi = T\mu^* \pi \); they exist under Assumption 2.1. We have \( x^* = T\mu^* \pi = T\nu^* \pi \) because \( Tx^* = T\pi x^* \) under Assumption 2.2. Lemma 2.3(i) and (ii) then imply that \( \mu^* \) and \( \nu^* \) are essentially proper, so by the result of [BT91], \( x^* \) is the optimal total reward function (optimal total cost function, respectively) of the single-player SSP problem for player II (player I, respectively) when player I takes policy \( \mu^* \) (player II takes policy \( \nu^* \), respectively). This optimality of \( x^* \) translates to:

\[
x(\mu^*, \pi_2) \leq x^* \leq x(\pi_1, \nu^*), \quad \forall \pi_1 \in \Pi_1, \pi_2 \in \Pi_2.
\]

We now prove \( x^* = x(\mu^*, \nu^*) \). Since \( \mu^* \) and \( \nu^* \) are essentially proper, by Lemma 2.2(i), \((\mu^*, \nu^*)\) is non-prolonging. Applying the result of [BT91] to the process induced by the non-prolonging pair \((\mu^*, \nu^*)\), which can be viewed as an uncontrollable SSP with a single (dummy) proper policy, we obtain that the total cost function under \((\mu^*, \nu^*)\) is the unique solution of the dynamic programming equation \( x = T\mu^* \pi x \). On the other hand, we have \( T\nu^* \pi x = x^* \) because \( x^* = T\pi x^* \leq T\nu^* \pi x^* \leq T\mu^* \pi x^* \) (cf. Eq. (2.4)). Therefore, \( x^* = x(\mu^*, \nu^*) \). Combining this with Eq. (2.6), we then have that \((\mu^*, \nu^*)\) is a pair of equilibrium (and optimal) policies for the two players and \( x^* \) is the value function of the game.

Next we consider value and policy iteration. Recall a well-known fact: if a monotone operator \( H : \mathbb{R}^n \to \mathbb{R}^n \) is nonexpansive with respect to the sup-norm (i.e., \( \|Hx - Hy\|_\infty \leq \|x - y\|_\infty \)) and has a unique fixed point \( \bar{x} \), then fixed point iterations \( H^k x \) converge to \( \bar{x} \) for any initial \( x \) (see e.g., [YB13b] Lemma 2.1 for a proof). The monotone mapping \( T \) is nonexpansive with respect to the sup-norm, and under Assumptions 2.1-2.3, it has a unique fixed point by Theorem 2.1. Therefore, \( x^* = x(\mu^*, \nu^*) \). Combining this with Eq. (2.6), we then have that \((\mu^*, \nu^*)\) is a pair of equilibrium (and optimal) policies for the two players and \( x^* \) is the value function of the game.

Policy iteration for each player starting with an essentially proper policy also converges under Assumptions 2.1-2.3. This is shown below. Since our SSP game model is symmetric for the two players, it suffices to discuss the case of player I. In policy iteration, starting from a policy \( \mu_0 \in D_1 \) that is essentially proper, we define recursively \( x_t \in \mathbb{R}^n \) and policy \( \mu_{t+1} \in D_1 \) by:

\[
x_t = T\mu_t x_t, \quad T\mu_{t+1} x_t = Tx_t, \quad t \geq 0.
\]

By induction, in the above \( x_t \) is well-defined (Lemma 2.1), \( \mu_{t+1} \) is well-defined under Assumption 2.1 and since \( x_t \geq T\mu_t x_t \), all \( \mu_{t+1} \) thus generated are essentially proper (Lemma 2.3). It can also be seen that \( \{x_t\} \) is a non-increasing sequence (using the fact that \( T\mu_{t+1} x_t \leq x_t \)). We summarize these results in the theorem below.

**Theorem 2.2** (Convergence of Value and Policy Iteration). Under Assumptions 2.1-2.3 with \( x^* \) being the value function of the game, the following holds:

(i) Convergence of value iteration: For any \( x \in \mathbb{R}^n \), \( \lim_{t \to \infty} T^t x = x^* \).

---

More precisely, the argument for \( x(\mu^*, \pi_2) \leq x^* \), \( \pi_2 \in \Pi_2 \), is the following. For the total reward SSP problem resulting from player I taking policy \( \mu^* \), consider the corresponding total cost problem with one-stage costs being \( -c_i(u, \pi) \). Then, by [BT91], \( -x^* \) is the optimal total cost function, and hence, for every state \( i \) and \( \pi_2 \in \Pi_2 \),

\[
-x^*_i \leq \liminf_{t \to \infty} E_{\mu^*, \pi_2} \left[ -\sum_{k=0}^{t} c_i(u_k, \pi_k) \mid i_0 = i \right] \\
= -\limsup_{t \to \infty} E_{\mu^*, \pi_2} \left[ \sum_{k=0}^{t} c_i(u_k, \pi_k) \mid i_0 = i \right] \leq -\liminf_{t \to \infty} E_{\mu^*, \pi_2} \left[ \sum_{k=0}^{t} c_i(u_k, \pi_k) \mid i_0 = i \right] = -x_i(\mu^*, \pi_2),
\]

which is \( x^* \geq x(\mu^*, \pi_2) \).
(ii) Convergence of policy iteration: Let \( \{x_t\} \) and \( \{\mu_t\} \) be defined by Eq. (2.7) with \( \mu_0 \) being essentially proper for player I. Then all \( \mu_t \) are essentially proper. Furthermore, \( \lim_{t \to \infty} x_t = x^* \), and any cluster point \( \mu_\infty \) of \( \{\mu_t\} \) is essentially proper and optimal for player I.

Proof. We prove the last statement in (ii); the other statements are already proved in the preceding discussion. We have the relation

\[
x_t \geq T x_t = T \mu_{t+1} x_t \geq x_{t+1}.
\]

(To see this, note that since \( T \mu_t x_t \geq T x_t \) [cf. Eq. (2.5)], we have \( x_t \geq T x_t = T \mu_{t+1} x_t \). Using the monotonicity of \( T \mu_{t+1} \), the fact that \( \mu_{t+1} \) is essentially proper, and Lemma 2.3, we then obtain \( T \mu_{t+1} x_t \geq x_{t+1} \).) Hence the sequence \( \{x_t\} \) is non-increasing. Since all \( \mu_t \) are essentially proper, by Lemma 2.2(ii), \( \{x_t\} \) is bounded below by \( \bar{x} \in \mathbb{R}^n \), the unique fixed point of \( T \nu \), where \( \nu \) is any policy of player II that is essentially proper. (By Lemma 2.3, such a policy \( \nu \) exists under Assumption 2.3.) Therefore, \( \{x_t\} \) converges to some \( x_\infty \in \mathbb{R}^n \). Using the relation \( x_t \geq T x_t \geq x_{t+1} \) and the continuity of \( T \), we obtain that \( x_\infty \geq T x_\infty \geq x_\infty \), i.e., \( x_\infty = T x_\infty \). Since \( x^* \) is the unique fixed point of \( T \) (Theorem 2.1), we have \( x_\infty = x^* \).

Let \( \mu_\infty \) be a cluster point of \( \{\mu_t\} \). Since every component of \( T \mu x \) is a lower semicontinuous function of \( (\mu, x) \) under Assumption 2.1, we obtain from the relation \( \mu \geq T \mu_{t+1} x_t \) and the convergence of \( \{x_t\} \) to \( x^* \) that \( x^* \geq T \mu_\infty x^* \). By Lemma 2.2(i), this implies that \( \mu_\infty \) is essentially proper. We also have, by Eq. (2.5), that \( T \mu_\infty x^* \geq T x^* = x^* \). Hence \( T \mu_\infty x^* = T x^* \) and by Theorem 2.1, \( \mu_\infty \) is an optimal policy for player I.

2.3 Further Remarks

The results we presented in this section bear close relations to those given in the earlier work [PB99] on SSP games. In what follows we make a detailed comparison of our model assumption with the formulation in [PB99], and we also discuss the scope and limitation of our model through a well-known example.

Patek and Bertsekas [PB99] formulated an SSP game model and derived optimality results similar to ours. The model conditions of [PB99, Assumption SSP] are stated in terms of deterministic Markov policies \( \pi_1, \pi_2 \) (instead of stationary policies) of the two players:

(i) There exists a policy \( \pi_1 \) of player I such that for all policies of player II, the termination state is reached w.p.1 for all initial states.

(ii) For every pair of policies \( (\pi_1, \pi_2) \) that is prolonging, the expected total cost of player I is infinite for at least one initial state \( i \), i.e., \( x_i(\pi_1, \pi_2) = +\infty \).

Instead of the essentially proper policies as we have defined, the well-behaved policies in their framework, which they call proper policies, are the ones for player I under which the game terminates no matter how player II plays. Under the above assumptions and continuity and regularity conditions, they obtained optimality results similar to Theorems 2.1 and 2.2 for their model. Their results ensure that player I has an optimal stationary proper policy and policy iteration for player I converges when starting from a proper policy. By contrast, our model formulation is centered on essentially proper policies, under which the game need not terminate for all policies of the other player, and our results ensure the existence of optimal stationary policies for both players within the class of essentially proper policies, as well as the convergence of policy iteration starting with such a policy.

Let us discuss more about the above model assumptions (i)-(ii) considered by [PB99]. Because of the non-stationarity of policy \( \pi_1 \), it is not immediate to see what implication assumption (i) has on the structure of the game. However, based on the results and analyses of [PB99], when assumption (ii) and other continuity/regularity conditions are in force, assumption (i) is equivalent to:
(i') There exists a policy $\bar{\mu} \in D_1$ of player I such that for all $\nu \in D_2$ of player II, $(\bar{\mu}, \nu)$ is non-prolonging.

 Assumptions (i') and (ii) clearly imply our Assumption 2.3 so our SSP game model covers a larger class of games. Assumptions (i)-(ii) or (i')-(ii) are also asymmetric for the two players, whereas Assumption 2.3 has a symmetric form.

 To see why Assumption 2.3 characterizes a much broader class of games than the model conditions (i)-(ii) of [PB99] do, we note two main restrictions in the latter conditions. First, assumption (i) overly favors player I by requiring that player I can terminate the game however player II plays, whereas assumption (ii) overly favors player II by requiring that a non-terminating game is always to the disadvantage of player I. Although these conditions seem natural for those applications in which player I is a “pursuer” and player II an “evader,” and the game is over when the pursuer achieves the goal of catching the evader (see [PB99, Sec. 5]), they are restrictive for games not of the pursuit-evasion type. Second, by imposing the condition in assumption (ii) on every pair of prolonging policies, assumptions (i)-(ii) effectively require that if the two players play only stationary policies, then against any given strategy of player II, player I will not be able to obtain strictly negative average cost ($-\infty$ total cost), for any initial state. Consider the implication of this for a finite state and control game, for example. (In a finite state and control game, $\bar{U}(i)$ and $\bar{V}(i)$ correspond to the distributions over the finite control sets at state $i$ under randomized stationary policies, and $\pi_1$ and $\pi_2$ correspond to randomized Markov policies.) Then the requirement imposed by assumptions (i)-(ii) just mentioned entails that against any given deterministic stationary policy of player II, player I cannot find a stationary policy to obtain an infinite amount of return by prolonging the game. This is a serious restriction in the model formulation of [PB99].

 To end this section, we discuss some examples of total cost zero-sum games that are excluded by our model assumptions. The following simple finite state and control game, due to Everett [Ever57], has no optimal policy for player II but has a value [KSS1]. The state space is $S_0 = \{0, 1\}$. At state 1, there are two controls $\{1, 2\}$ for each player, and when player I applies control $u$ and player II control $v$, the system transitions to state 0 with cost 1 if $u = v$, transitions to state 0 with cost 0 if $u = 1, v = 2$, and transitions to state 1 with cost 0 if $u = 2, v = 1$. (Here $\bar{U}(1), \bar{V}(1)$ are given by the set of probability distributions on $\{1, 2\}$.) The value of the game for state 1 is 1. This example violates Assumption 2.3(iii) because the pair of policies with player I applying control 2 and player II control 1 at state 1, is prolonging but incurs zero total cost.

 It is worth to mention that in this example, although not every player has an optimal policy, the dynamic programming equation $x = Tx$ of the game does have a unique solution, which is the value of the game [KSS1]. It is also easy to construct examples where the game has a value, both players have stationary optimal policies, and the dynamic programming equation has a unique solution, but there exists a prolonging pair of stationary optimal policies (with zero average cost) so that the game will be excluded by our model assumption. Here is the simplest such example: let the state and control spaces be as in the preceding example; let all one-stage costs be zero; and at state 1, let the system transit to state 0 if either player applies control 1, and let the system stay at state 1 otherwise. In this game, all policies are optimal, the value of the game is zero, and it is the unique solution of the dynamic programming equation. This illustrates that not all games with nice optimality properties are included in the class of games satisfying our model assumptions.

## 3 Q-Learning for Finite-Space SSP Games

Starting with this section, we will focus on total cost zero-sum games with a finite state and control space and analyze the convergence of a model-free, stochastic approximation-based algorithm, Q-learning, for solving these games. In this section, first, the SSP game model introduced in Section 2 will be specialized to the finite-space game context, to provide a finite-space SSP game model that
has desirable optimality properties for applying the Q-learning algorithm. The Q-learning algorithm will then be introduced, along with the convergence results we have obtained. The major proofs for these results will be given in the next section.

3.1 Finite-Space SSP Games

Consider a finite state and control two-player zero-sum game. The state space is $S_0 = S \cup \{0\}$ as before, where 0 is the cost-free termination state. At state $i \in S$, each player has a finite set of feasible controls, denoted by $U(i)$, $V(i)$ for player I, player II, respectively. The rules of the game and the objectives of the two players are as described in Section 2.4. However, with apologies to the readers, we will use some different notation to make it conforming to standard notation in the Q-learning literature. In particular, for each pair of controls $(u, v) \in U(i) \times V(i)$, let $p_{ij}(u, v)$ be the probability of transition from state $i$ to $j \in S_0$, let $\bar{g}(i, u, v, j)$ be the corresponding compact-control game where:

$$\nu_{ij} = \sum_{j \in S} p_{ij}(u, v) \bar{g}(i, u, v, j)$$

let $g(i, u, v, j)$ denote the expected one-stage cost at state $i$ with controls $(u, v)$. At each time $t$, every player may use the information of the current state $i_t$ and the history of the game, including the past states $\{i_k, k < t\}$ and past controls $\{u_k, v_k, k < t\}$ of both players, to decide which control to apply. When player I adopts policy $\pi_1$ and player II $\pi_2$, we write the total cost of player I starting from initial state $i$ as $J(i; \pi_1, \pi_2)$, i.e.,

$$J(i; \pi_1, \pi_2) = \lim_{t \to \infty} \mathbb{E}_{\pi_1, \pi_2} \left[ \sum_{k=0}^{t} g(i_k, u_k, v_k, i_{k+1}) \mid i_0 = i \right].$$

(Since we will deal with asynchronous iterative algorithms in this section, we find the notation $J(i; \pi_1, \pi_2)$ more convenient than the notation $x_t(\pi_1, \pi_2)$ of Section 2 thus reserving subscripts for iteration indices.) As before, for every state $i$, the two players’ goals are:

player I: minimize $\sup_{\pi_1 \in \Pi_1} J(i; \pi_1, \pi_2)$, player II: maximize $\inf_{\pi_2 \in \Pi_2} J(i; \pi_1, \pi_2)$.

Of particular importance are stationary randomized policies. For each state $i \in S$, let $\bar{U}(i) = \mathcal{P}(U(i))$ and $\bar{V}(i) = \mathcal{P}(V(i))$ denote the set of probability distributions on $U(i)$ and $V(i)$, respectively, which are the randomized decision rules of the two players for state $i$. A stationary randomized policy of a player takes the form,

for player I: $\mu = \{\mu(\cdot \mid i) \mid i \in S\}$, where $\mu(\cdot \mid i) \in \bar{U}(i)$,

for player II: $\nu = \{\nu(\cdot \mid i) \mid i \in S\}$, where $\nu(\cdot \mid i) \in \bar{V}(i)$.

With such a policy $\mu$ ($\nu$, resp.), at state $i$, player I (player II, resp.) takes control $u$ ($v$, resp.) with probability $\mu(u \mid i)$ ($\nu(v \mid i)$, resp.). We denote the set of stationary randomized policies of player I and player II by $\Pi_{1,SR}$ and $\Pi_{2,SR}$, respectively.

We can relate the above finite-space game to a finite-state compact-control game considered in Section 2, where the compact control sets correspond to the sets of randomized decision rules of each player in the present context. In particular, in the framework of Section 2 consider the corresponding compact-control game where:

(a) The compact control sets at state $i$ for the two players are given by the sets $\bar{U}(i), \bar{V}(i)$ defined above. For a pair $(\rho, \sigma) \in \bar{U}(i) \times \bar{V}(i)$, the probability of transition to state $j$ is given by $\sum_{u \in U(i)} \sum_{v \in V(i)} \rho(u) \sigma(v) p_{ij}(u, v)$, whereas the expected one-stage cost is given by $\sum_{u \in U(i)} \sum_{v \in V(i)} \rho(u) \sigma(v) g(i, u, v)$. These transition probabilities and one-stage costs satisfy the continuity/semi-continuity conditions in Assumption 2.1.

More generally, the transition cost can also depend on some additional stochastic disturbance $\omega$ and take the form $\bar{g}(i, u, v, j, \omega)$. Our analysis of Q-learning applies to such type of random transition costs provided that they have bounded variance, but for notational simplicity, we do not introduce them in the paper.
The same argument applied to player II. We have

\[ J(i) = (TJ)(i) := \inf_{\rho \in U(i)} \sup_{\sigma \in V(i)} \sum_{u \in U(i)} \sum_{v \in V(i)} \rho(u)\sigma(v) \left( g(i, u, v) + \sum_{j \in S} p_{ij}(u, v)J(j) \right), \quad \forall i \in S. \]

(3.1)

The SSP game model given in Assumption 2.3 then translates to the following model condition on finite-space games:

**Assumption 3.1 (Finite-Space SSP Game Model).** Assumption 2.3 holds for \( D_1 = \Pi_{1,SR} \) and \( D_2 = \Pi_{2,SR} \). That is, (i) player I (player II) has a stationary randomized policy under which the player’s total cost (reward) is less than +∞ (greater than −∞) no matter what stationary randomized policy the other player takes; and (ii) under any prolonging pair of stationary randomized policies of the two players, there is some initial state for which either the total cost for player I is +∞ or the total reward for player II is −∞.

Under Assumption 3.1, Theorems 2.1 and 2.2 apply to finite-space games through their associated compact-control games just described. In particular, we obtain from Theorem 2.1:

**Proposition 3.1 (Optimality Properties of Finite-Space SSP Games).** For a finite-space SSP game satisfying Assumption 3.1, there exist equilibrium policies \((\mu^*, \nu^*) \in \Pi_{1,SR} \times \Pi_{2,SR}\) for the two players, i.e.,

\[ J(i; \mu^*, \pi_2) \leq J(i; \mu^*, \nu^*) \leq J(i; \pi_1, \nu^*), \quad \forall \pi_1 \in \Pi_1, \pi_2 \in \Pi_2, i \in S. \]

(3.2)

In translating Theorem 2.1 into Proposition 3.1, there is a small technical detail that we need to mention: the policy spaces \( \Pi_1, \Pi_2 \) in the finite-space game are not the policy spaces in the corresponding compact-control game. Let us denote the latter sets by \( \bar{\Pi}_1, \bar{\Pi}_2 \) for the two players respectively. In general a history-dependent policy in \( \Pi_1 \) or \( \Pi_2 \) does not necessarily lie in \( \Pi_1 \) or \( \Pi_2 \). This is because the player in the finite-space game does not observe the randomized decision rules that the other player took in the past, and therefore cannot make control decisions based on that information, whereas the player in the corresponding compact-control game can use that information for control. However, Markov policies, in particular stationary policies, for the finite-space game are also policies for the other game. We use this fact together with a standard Markovian property in MDP to obtain the desired results for the finite-space game. For example, we can prove Eq. (3.2) as follows. By a direct application of Theorem 2.7 to the compact-control game, there exist \((\mu^*, \nu^*) \in \Pi_{1,SR} \times \Pi_{2,SR}\) with

\[ J(i; \mu^*, \pi_2) \leq J(i; \mu^*, \nu^*) \leq J(i; \pi_1, \nu^*), \quad \forall \pi_1 \in \Pi_1, \pi_2 \in \Pi_2, i \in S. \]

To obtain Eq. (3.2) from this inequality, consider first the total cost \( J(i; \pi_1, \nu^*) \) for any given state \( i \) and policy \( \pi_1 \in \Pi_1 \) in the finite-space game. Because the state evolves in a Markovian way when player II plays the stationary policy \( \nu^* \), one can construct a randomized Markov policy \( \bar{\pi}_1 \) such that \( J(i; \bar{\pi}_1, \nu^*) = J(i; \pi_1, \nu^*) \) (such construction is well-known in the MDP theory). Since a randomized Markov policy of player I lies in the intersection \( \Pi_1 \cap \Pi_1 \), we have \( J(i; \mu^*, \nu^*) \leq J(i; \bar{\pi}_1, \nu^*) \) by the preceding inequality, and consequently, \( J(i; \mu^*, \nu^*) \leq J(i; \pi_1, \nu^*) \) for any \( \pi_1 \in \Pi_1 \) and \( i \in S \). This proves the second half of the desired inequality (3.2). The other half of (3.2) follows from the same argument applied to player II.
The value function of the game, given by \( J^*(\cdot) = J(\cdot; \mu^*, \nu^*) \), is the unique solution of the dynamic programming equation \( J = TJ \) given by (3.1). Moreover, any \( \mu^* \in \Pi_{1,SR} \), \( \nu^* \in \Pi_{2,SR} \) such that \( T_{\mu^*}J^* = TJ^* \), \( T_{\nu^*}J^* = TJ^* \) are optimal policies of player I and player II, respectively, and they are essentially proper, with the pair \((\mu^*, \nu^*)\) forming a non-prolonging pair of equilibrium policies.

**Remark 3.1.** In a sequential game, only one player can move at each time and whose turn to move depends on the current state (see e.g., [BT96, Section 7.2]). Equivalently, at each state, one of the two players has a singleton control set. Then, from the definition of \( T \) and \( \tilde{T} \) [cf. Eq. (3.1)] it follows that for a sequential SSP game satisfying Assumption 3.1 both players have stationary deterministic equilibriums policies. It also follows that for sequential games, we may replace the sets \( \Pi_{1,SR} \) and \( \Pi_{2,SR} \) in Assumption 3.1 and Proposition 3.1 by the sets of stationary deterministic policies of the two players.

From Theorem 2.2 we obtain convergence of value and policy iteration for the finite-space game under Assumption 3.1. We will not focus on these algorithms in the rest of this paper, however. Instead, we will focus on a model-free algorithm called Q-learning, for computing the value function of the game. The algorithm is useful when the transition probabilities and expected one-stage costs are unknown or when the model is too complicated to have these parameters written down explicitly, but random transitions and transition costs can be observed or generated by a simulator. The Q-learning algorithm may be viewed as a stochastic value iteration algorithm. Standard value iteration, however, computes \( T J \) for some vector \( J \) at each iteration. It would be difficult to do so in the model-free context, with only a few observations of state transitions, as can be seen from the expression of \( T J \) in Eq. (3.1). The Q-learning algorithm will work not with the cost vector \( J \) but with the so-called Q-factors and an associated dynamic programming equation, which is equivalent to the dynamic equation \( J = TJ \) by a change of variable (from \( J \) to Q-factors). To prepare for the study of the Q-learning algorithm, let us explain this equation now.

**Q-Factors and the Associated Dynamic Programming Equation**

Let \( R = \{(i, u, v) \mid i \in S, u \in U(i), v \in V(i)\} \) be the state-and-control space. In the dynamic programming equation (3.1), with which we repeat here:

\[
J(i) = (TJ)(i) = \inf_{\rho \in U(i)} \sup_{\sigma \in V(i)} \sum_{u \in U(i)} \sum_{v \in V(i)} \rho(u)\sigma(v) \left(g(i, u, v) + \sum_{j \in S} p_{ij}(u, v)J(j)\right), \quad \forall i \in S,
\]

let us make a change of variable from \( J \) to \( Q = \{Q(i, u, v) \mid (i, u, v) \in R\} \) by letting

\[
Q(i, u, v) = g(i, u, v) + \sum_{j \in S} p_{ij}(u, v)J(j), \quad (i, u, v) \in R.
\]

This gives an equation in terms \( Q \): for all \((i, u, v) \in R\),

\[
Q(i, u, v) = g(i, u, v) + \sum_{j \in S} p_{ij}(u, v)(TJ)(j) = g(i, u, v) + \sum_{j \in S} p_{ij}(u, v) \inf_{\rho \in U(j)} \sup_{\sigma \in V(j)} \sum_{u' \in U(j)} \sum_{v' \in V(j)} \rho(u')\sigma(v')Q(j, u', v'). \tag{3.3}
\]

To simplify notation, we define the shorthand notation

\[
Q(i, \rho, \sigma) = \sum_{u \in U(i)} \sum_{v \in V(i)} \rho(u)\sigma(v)Q(i, u, v) \tag{3.4}
\]
for a given vector $Q$ and randomized decision rules $\rho \in U(i), \sigma \in V(i)$ for a state $i$. Then Eq. (3.5) can be expressed concisely as

$$Q = FQ \quad \text{or} \quad Q(i, u, v) = (FQ)(i, u, v), \quad \forall (i, u, v) \in R,$$

where the operator $F : \mathbb{R}^{|R|} \rightarrow \mathbb{R}^{|R|}$ is given by

$$(FQ)(i, u, v) := g(i, u, v) + \sum_{j \in S} p_{ij}(u, v) \inf_{\rho \in U(j)} \sup_{\sigma \in V(j)} Q(j, \rho, \sigma), \quad (i, u, v) \in R. \quad (3.6)$$

We refer to the components of $Q$ as Q-factors. Equation $Q = FQ$ given by (3.5) is the dynamic programming equation for Q-factors. Since it is obtained from $J = TJ$ by a change of a variable, any solution of $J = TJ$ gives us a solution of $Q = FQ$. Conversely, if in the equation $Q = FQ$ we change the variable $Q$ to $J$ by letting

$$J(i) = \inf_{\rho \in U(i)} \sup_{\sigma \in V(i)} Q(i, \rho, \sigma), \quad \forall i \in S,$$

then by a direct calculation, we get back the equation $J = TJ$. Hence any solution of $Q = FQ$ gives us a solution of $J = TJ$. Furthermore, it can be verified using the definition of $F$ and $T$ that there is a one-to-one correspondence between the solutions of these two dynamic programming equations. Using these facts, some optimality properties given in Prop. 3.1 can be stated in terms of Q-factors as follows:

**Corollary 3.1** (Optimality Properties of Finite-Space SSP Games in terms of Q-factors). For a finite-space SSP game satisfying Assumption 3.1, the dynamic programming equation (3.5) has a unique solution $Q^*$, which relates to the value function $J^*$ of the game by

$$Q^*(i, u, v) = g(i, u, v) + \sum_{j \in S} p_{ij}(u, v)J^*(j), \quad \forall (i, u, v) \in R,$$

$$J^*(i) = \inf_{\rho \in U(i)} \sup_{\sigma \in V(i)} Q^*(i, \rho, \sigma), \quad \forall i \in S.$$

Any stationary policies $\mu^* \in \Pi_{1,SR}, \nu^* \in \Pi_{2,SR}$ such that for every state $i$,

$$\mu^*(\cdot \mid i) \in \arg\min_{\rho \in U(i)} \sup_{\sigma \in V(i)} Q^*(i, \rho, \sigma), \quad \nu^*(\cdot \mid i) \in \arg\max_{\sigma \in V(i)} \inf_{\rho \in U(i)} Q^*(i, \rho, \sigma),$$

are optimal policies for the two players.

As Cor. 3.1 shows, for an SSP game satisfying Assumption 3.1, if we know $Q^*$, we can use it to compute the value function of the game and optimal policies of the two players, by solving for each state a matrix game defined by $Q^*$: $\min_{\rho \in U(i), \sigma \in V(i)} \max_{\rho \in U(i), \sigma \in V(i)} Q^*(i, \rho, \sigma)$. These matrix game problems do not involve the parameters of the SSP game, which can be unknown in the learning context.

Corollary 3.1 also shows that under Assumption 3.1 $Q^*$ is the unique fixed point of the dynamic programming operator $F$, and therefore, since $F$ is also monotone and nonexpansive with respect to the sup-norm $\| \cdot \|_\infty$ by definition, the fixed point iteration $Q_{t+1} = FQ_t$ converges to $Q^*$ for any initial $Q_0$. These properties are important for applying the Q-learning algorithm to compute $Q^*$.

### 3.2 Q-Learning for SSP Games

The Q-learning algorithm is an asynchronous stochastic iterative algorithm, and as mentioned earlier, it does not require the knowledge of the model parameters such as transition probabilities
and expected one-stage costs. Instead, its computation is based on random state transitions and transition costs, which may be generated by a simulator or observed in a real learning environment.

We consider using Q-learning to compute the function $Q^*$ for a finite-space SSP game satisfying Assumption 3.1. Intuitively, one may view the algorithm as a stochastic version of damped fixed point iterations with the mapping $F$, i.e., iterations of the form $(1 - \gamma)Q + \gamma FQ$ for some stepsize parameter $\gamma$. The algorithm generates iteratively a sequence of Q-factor vectors, $\{Q_t\}$. Our main result is a proof that this sequence converges to $Q^*$ w.p.1 in a fairly general totally asynchronous computation setting.

To describe the algorithm, first recall that

$$(FQ)(i, u, v) = g(i, u, v) + \sum_{j \in S} p_{ij}(u, v) \inf_{\rho \in U(j)} \sup_{\sigma \in V(j)} Q(j, \rho, \sigma)$$

[cf. Eqs. (3.6), (3.4)], so a damped fixed point iteration $Q_{t+1} = (1 - \gamma)Q_t + \gamma FQ_t$ will set the $(i, u, v)$-th component of $Q_{t+1}$ to be

$$Q_{t+1}(i, u, v) = (1 - \gamma)Q_t(i, u, v) + \gamma \left(g(i, u, v) + \sum_{j \in S} p_{ij}(u, v) \inf_{\rho \in U(j)} \sup_{\sigma \in V(j)} Q(j, \rho, \sigma)\right).$$

The Q-learning algorithm we describe next differs from the above iteration in several ways:

(i) It is an asynchronous algorithm. At each iteration, it updates only a chosen subset of Q-factor components, keeping the rest unchanged.

(ii) Its computation can be distributed among multiple processors. Each Q-factor component can be updated by a separate processor, for example, and communication delays are taken into account by allowing a processor to use outdated information in computation. In particular, for updating the $(i, u, v)$-th component at iteration $t$, the algorithm can use the Q-factor component $Q_{\tau}(j, \tilde{u}, \tilde{v})$ computed at some iteration $\tau \leq t$, where $\tau$ can depend on both $(i, u, v)$ and $(j, \tilde{u}, \tilde{v})$, reflecting the communication delay between the two associated processors. In the algorithm, we will write these $\tau$ variables as $\tau_{\ell}(t)$, for every pair of state-control triplets $\ell = (i, u, v), \tilde{\ell} = (j, \tilde{u}, \tilde{v}) \in R$. For each $\ell = (i, u, v) \in R$, we will use the shorthand notation $Q_t^{(\ell)}$ to denote the Q-factor vector whose $(j, \tilde{u}, \tilde{v})$-th component is given by:

$$Q_t^{(\ell)}(j, \tilde{u}, \tilde{v}) = Q_{\tau_{\ell}(t)}(j, \tilde{u}, \tilde{v}) \quad \text{with} \quad \tilde{\ell} = (j, \tilde{u}, \tilde{v}) \in R. \tag{3.7}$$

For $\ell = (i, u, v)$, we can view $Q_t^{(\ell)}$ as the “local information” that the $\ell$th processor uses for updating $Q_{t+1}(i, u, v)$.

(iii) It is a model-free, stochastic approximation-based algorithm. Compared with the damped fixed point iteration $Q_{t+1}(i, u, v) = (1 - \gamma)Q_t(i, u, v) + \gamma (FQ_t^{(\ell)})(i, u, v)$ using possibly “outdated” information as just discussed, the Q-learning iterate for $Q_{t+1}(i, u, v)$ uses, in place of $(FQ_t^{(\ell)})(i, u, v)$, an unbiased estimate of $(FQ_t^{(\ell)})(i, u, v)$ obtained through sampling state transitions randomly.

Let us describe now the Q-learning algorithm. The algorithm generates recursively a sequence $\{Q_t\}$ of Q-factor vectors. At each iteration, it generates random state transitions, and the termination state $0$ and the zero total cost at that state appear explicitly in the calculation. For notational convenience, let us define for state $0$, the dummy control sets $U(0) = V(0) = \{0\}$ with

---

Note that the behavior of the Q-learning algorithm in practice is, however, much more complex than suggested by this simple view (in the context of MDP, its behavior can sometimes resemble policy iteration, for example). Such complexity can be attributed in part to various coordination schemes one can use with asynchronous and distributed computation. This subject is beyond the scope of this paper, however.
\[ \hat{V}(0) = \hat{V}(0) = \mathcal{P}(\{0\}) \], and treat Q-factors as \((|R| + 1)\)-dimensional vectors with \(Q(0, 0, 0) = 0\). It will be taken for granted that \(Q_t(0, 0, 0) = 0\) for all \(t\) and the variables \(\tau_{t\ell}(\hat{\ell})\) for communications delays between \(\ell \in R\) and \(\hat{\ell} = (0, 0, 0)\) are (arbitrarily) defined. Given \(\{Q_\tau, \tau \leq t\}\), the \(t\)th iteration of the algorithm computes \(Q_{t+1}\) as follows.

**Q-Learning Algorithm (\(t\)th iteration)**

For each state-control triplet \(\ell = (i, u, v) \in R\):

(a) Let \(\gamma_{t\ell} \in [0, 1]\) be a stepsize parameter. For each \(\ell \in R\), let \(\tau_{t\ell}(t) \leq t\) be a nonnegative integer.

(b) Generate a random transition from state \(i\) with control \((u, v)\), and denote the successor state by \(j^\ell_t\) (here \(j^\ell_t \in S_o\)). With \(s\) being a shorthand for the state \(j^\ell_t\), let

\[
Q_{t+1}(i, u, v) = \left(1 - \gamma_{t\ell}\right)Q_t(i, u, v) + \gamma_{t\ell} \left(\hat{g}(i, u, v, s) + \inf_{\rho \in U(s)} \sup_{\sigma \in V(s)} Q^\ell(s, \rho, \sigma)\right).
\]

(3.8)

Here for \(s \neq 0\) (i.e., \(s\) is not the termination state), \(Q^\ell(s, \rho, \sigma)\) is a shorthand notation for the weighted average of Q-factors,

\[
\sum_{\tilde{u} \in U(s)} \sum_{\tilde{v} \in V(s)} \rho(\tilde{u})\sigma(\tilde{v})Q^\ell(s, \tilde{u}, \tilde{v}),
\]

with \(Q^\ell(s, \rho, \sigma)\) being the Q-factor vector given by Eq. (3.7). For \(s = 0\), \(Q_t(0, 0, 0) = 0\) [which is also consistent with the preceding expression when we extend the definition in Eq. (3.7) to include \(\ell = (0, 0, 0)\)].

We note that the stepsize variables specify implicitly the subset of Q-factor components to be updated at iteration \(t\). If \(\gamma_{t\ell} = 0\), then \(Q_{t+1}(\ell) = Q_t(\ell)\) and no computation is actually needed to carry out step (b). The components with positive stepsizes, \(\{\ell \in R \mid \gamma_{t\ell} > 0\}\), are those for which the corresponding Q-factors are selected for an update.

The variables appearing in the Q-learning algorithm will be regarded as random variables on a common probability space \((\Omega, \mathcal{F}, \mathbf{P})\). We require them to satisfy the following standard conditions for asynchronous Q-learning (cf. [18394]). (In fact, without these conditions, the algorithm as just described is imprecise.) Let \(\{\mathcal{F}_t\}\) be an increasing sequence of sub-\(\sigma\)-fields of \(\mathcal{F}\). (They represent the histories of the algorithm up to certain times.)

**Assumption 3.2 (Algorithmic Conditions).**

(i) \(Q_0\) is \(\mathcal{F}_0\)-measurable.

(ii) For every \(\ell, \hat{\ell} \in R\) and \(t \geq 0\), \(\gamma_{t\ell}\) and \(\gamma_{t\hat{\ell}}\) are \(\mathcal{F}_t\)-measurable.

(iii) For every \(\ell = (i, u, v) \in R\) and \(t \geq 0\), \(j^\ell_t\) is \(\mathcal{F}_{t+1}\)-measurable and

\[
\mathbf{P}(j^\ell_t = j \mid \mathcal{F}_t) = p_{ij}(u, v), \quad j \in S_o.
\]

(3.9)

(iv) With probability 1,

\[
\lim_{t \to \infty} \tau_{t\ell}(t) = \infty, \quad \forall \ell, \hat{\ell} \in R.
\]

(3.10)

(v) With probability 1,

\[
\sum_{\ell \geq 0} \gamma_{t\ell} = \infty, \quad \sum_{\ell \geq 0} \gamma_{t\ell}^2 < \infty, \quad \forall \ell \in R.
\]

(3.11)
Conditions (i)-(iii) are on the probabilistic dependence relations between the variables. They are naturally satisfied by the Q-learning algorithm in practice, when at each iteration, the values of stepsizes and communication delays are chosen before the random successor states are generated. Condition (iv) is on the variables related to communication delays: it ensures that outdated information will eventually be purged by the algorithm, so it is a minimal requirement for totally asynchronous computation. Condition (v) is a standard stepsize condition. It implies that every Q-factor component is updated infinitely often, which is certainly indispensable for the Q-learning algorithm to find \( Q^* \) in the limit.

We have the following results regarding the convergence of the Q-learning algorithm given above.

**Theorem 3.1** (Boundedness of Q-Learning Iterates). Consider a finite-space SSP game satisfying Assumption 3.1. Then under Assumptions 3.2(i)-(iii) and (v), for any given initial \( Q_0 \), the sequence \( \{Q_t\} \) generated by the Q-learning algorithm (3.8) is bounded w.p.1.

**Theorem 3.2** (Convergence of Q-Learning). Consider a finite-space SSP game satisfying Assumption 3.1. Then under Assumption 3.2, for any given initial \( Q_0 \), the sequence \( \{Q_t\} \) generated by the Q-learning algorithm (3.8) converges w.p.1 to the unique solution \( Q^* \) of the equation \( Q = FQ \).

Theorem 3.1 on the boundedness of \( \{Q_t\} \) is our main result. Its proof will be the subject of the next section. Assuming it has been proved, the convergence of Q-learning stated in Theorem 3.2 follows by combining the boundedness result with a convergence theorem of Tsitsiklis [Tsi94]. We give this proof below.

**Proof of Theorem 3.2.** To analyze the convergence of the Q-learning iterates \( Q_t \), we write them in a form that is standard for stochastic approximation-based analysis. For every \( \ell = (i, u, v) \in R \) and every \( t \geq 0 \), we express the iteration (3.8) equivalently as

\[
Q_{t+1}(i, u, v) = (1 - \gamma_{t,\ell})Q_t(i, u, v) + \gamma_{t,\ell}(FQ_t^{(\ell)})(i, u, v) + \gamma_{t,\ell}w_{t,\ell},
\]

(3.12)

where \( w_{t,\ell} \) is a noise term given by

\[
w_{t,\ell} = \hat{g}(i, u, v, s) + \inf_{\rho \in U(s)} \sup_{\sigma \in V(s)} Q_t^{(\ell)}(s, \rho, \sigma) - (FQ_t^{(\ell)})(i, u, v),
\]

and \( s \) is a shorthand notation for the random successor state \( j_\ell^t \). Using Eq. (3.9) and the definition of the mapping \( F \) [cf. Eq. (3.6)], direct calculation shows that the noise terms in the iteration (3.12) satisfy that for every \( \ell \in R \) and \( t \geq 0 \),

\[
E[w_{t,\ell} \mid \mathcal{F}_t] = 0, \quad \text{w.p.1},
\]

and there exist deterministic constants \( A \) and \( B \), independent of \( \ell \) and \( t \), such that

\[
E[w_{t,\ell}^2 \mid \mathcal{F}_t] \leq A + B \max_{\ell \in R} \max_{\tau \leq t} |Q_\tau(\ell)|^2, \quad \text{w.p.1}.
\]

Then, since under Assumption 3.4, \( F \) has a unique fixed point \( Q^* \) and is monotone and nonexpansive with respect to \( \| \cdot \|_\infty \), a convergence theorem of Tsitsiklis [Tsi94, Theorem 2] applies and shows that \( \{Q_t\} \) converges to \( Q^* \) w.p.1, provided that \( \{Q_t\} \) is bounded w.p.1. The desired convergence result then follows from Theorem 3.1. \( \square \)

**Remark 3.2.** We have set the stepsizes \( \gamma_{t,\ell} \leq 1 \) in this paper. Theorems 3.1, 3.2 actually hold without this restriction, but in order to handle the general case of positive, possibly unbounded stepsizes, additional technical arguments are needed in the proofs, and such arguments can be found in the papers [YB13a, YB13b]. To avoid the technical complication and repetition, in this paper we choose not to focus on general stepsizes.
Remark 3.3. As mentioned in the introduction section, using the O.D.E.-based analysis, Abounadi, Bertsekas and Borkar \cite{ABB02} established convergence for a class of asynchronous stochastic approximation algorithms involving nonexpansive mappings, and their results can be applied to the Q-learning algorithm for SSP games we consider. However, their asynchronous computation framework differs from the totally asynchronous computation framework we consider here. A chief assumption in their framework is that all the components are updated comparatively often in the sense that \( \liminf_{t \to \infty} m(t, \ell)/t > 0 \) for all components \( \ell \), where \( m(t, \ell) \) is the number of times the \( \ell \)-th component has been updated up to time \( t \). (See also the related asynchronous schemes and their analyses in \cite{Bor98, BM00}, \cite[Chap. 7]{Bor08}.) If this and some other conditions on the stepsizes and communication delays are assumed to hold, the convergence result of \cite{ABB02} when applied in our context would lead to the conclusion that \( Q_t \) tracks the scaled O.D.E. \( \dot{Q} = \frac{1}{n} (FQ - Q) \). By comparison, the totally asynchronous Q-learning algorithm considered here is generally not to be expected to have such kind of behavior, since it does not restrict how often a component should be selected for update. Correspondingly, the boundedness and convergence analyses of the algorithm for the totally asynchronous case also differ significantly from the O.D.E.-based analyses in the aforementioned works.

4 Boundedness of Q-Learning Iterates

In this section we prove Theorem 3.1 on the boundedness of Q-learning iterates for a finite-space SSP game satisfying Assumption 3.1. The proof is long and uses a line of analysis devised earlier for bounding Q-learning iterates in single-player SSP problems (Yu and Bertsekas \cite{YB13a}). After the proof of Theorem 3.1 which takes up Section 4.1, we include in Section 4.2 a short boundedness proof for a special case where the assumption on the game model is more restrictive than Assumption 3.1 and the boundedness analysis is based on a contraction argument.

4.1 Boundedness Analysis for the General Case

In this subsection, we prove the boundedness of Q-learning iterates stated in Theorem 3.1. Assumption 3.1 implies that there exist a policy \( \bar{\mu} \in \Pi_{1,SR} \) of player I and a policy \( \bar{\nu} \in \Pi_{2,SR} \) of player II that are essentially proper (Lemma 2.3). We will prove the lower boundedness of \( \{Q_t\} \) by using the essential properness property of the policy \( \bar{\nu} \) and by using the implications of this property on the single-player SSP problem associated with the policy \( \bar{\nu} \). Due to symmetry, the same proof will also establish that \( \{-Q_t\} \) is bounded above w.p.1, by applying an identical argument to \( \{-Q_t\} \) and using the essential properness property of the policy \( \bar{\mu} \).

The proof consists of several steps, given in separate subsections. The main idea of the proof, reflected in the titles of these subsections, can be outlined as follows:

1. We relate \( \{Q_t\} \) to a sequence \( \{\hat{Q}_t\} \) of iterates that resembles Q-learning in the single-player SSP problem associated with the policy \( \bar{\nu} \). We show that lower boundedness of \( \{\hat{Q}_t\} \) implies lower boundedness of \( \{Q_t\} \). (See Section 4.1.1)

2. For any given positive scalar \( \delta \), we construct an auxiliary sequence \( \{\tilde{Q}_t\} \) such that (i) it is lower bounded w.p.1 if and only if \( \{Q_t\} \) is lower bounded w.p.1, and (ii) each component of \( \tilde{Q}_t \) can be interpreted as the total cost of some policy in a time-inhomogeneous SSP problem in the “\( \delta \)-neighborhood” of the single-player SSP problem associated with the policy \( \bar{\nu} \). (See Sections 4.1.2-4.1.5) These are the key steps of our proof.

3. We show that when \( \delta \) is sufficiently small, the optimal total costs of all the single-player SSP problems in the aforementioned “\( \delta \)-neighborhood” can be bounded uniformly from below, and hence the auxiliary sequence \( \{\hat{Q}_t\} \) is bounded below w.p.1. (See Section 4.1.6) This leads to
The desired conclusion that \( \{ \hat{Q}_t \} \) and hence \( \{ Q_t \} \) are bounded below w.p.1, completing the proof.

The auxiliary sequence-based arguments we use in this proof are first used in the boundedness analysis of Q-learning for single-player SSP problems \[ \text{YB13a}. \]

4.1.1 Relate \( \{ Q_t \} \) to Q-learning type iterations in a single-player SSP problem

To facilitate the analysis, we first reduce the question of lower boundedness of \( \{ Q_t \} \) to the question of lower boundedness of another process \( \{ \hat{Q}_t \} \), which is defined on the same probability space as \( \{ Q_t \} \). The advantage of working with \( \{ \hat{Q}_t \} \) is that we can relate it to Q-learning like iterations for a single-player SSP that satisfies the SSP Model Assumption.

Let \( \tilde{\nu} \in \Pi_{2,\text{all}} \) be an essentially proper policy of player II; the existence of such a policy is ensured by Lemma 2.3 under Assumption 3.1. To simplify notation, denote \( \hat{\nu}_i = \hat{\nu}(\cdot | i) \) for every \( i \in S_o \) [note \( \hat{\nu}_i \in \mathcal{P}(V(i)) \)]. We define an iteration similar to the Q-learning iteration \( 3.8 \), using the same random variables (i.e., \( \gamma_{t,\ell}, j_t^\ell \) and \( \tau_{t,\ell}(t, \ell, \ell) \)) that appear in the Q-learning iteration \( 3.8 \). In particular, let \( \hat{Q}_0 = Q_0 \) and for \( t \geq 0 \) and for every \( \ell = (i, u, v) \in R \), let

\[
\hat{Q}_{t+1}(i, u, v) = (1 - \gamma_{t,\ell}) \hat{Q}_t(i, u, v) + \gamma_{t,\ell} \left( \hat{g}(i, u, v, s) + \inf_{\rho \in \tilde{U}(s)} \hat{Q}_t^\ell(s, \rho, \tilde{\nu}_s) \right), \tag{4.1}
\]

where \( s \) is a shorthand for the successor state \( j_t^\ell \), and the expression \( \hat{Q}_t^\ell(s, \rho, \tilde{\nu}_s) \) denotes a weighted average of Q-factors given by

\[
\hat{Q}_t^\ell(s, \rho, \tilde{\nu}_s) = \sum_{\tilde{u} \in \tilde{U}(s)} \sum_{\tilde{v} \in V(s)} \rho(\tilde{u})\tilde{\nu}_s(\tilde{v})\hat{Q}_t^\ell(s, \tilde{u}, \tilde{v})
\]

with \( \hat{Q}_t^\ell \) being the vector whose \( \ell \)th component for \( \ell \in R \cup \{(0,0,0)\} \) is given by \( \hat{Q}_t^\ell(\tilde{\ell}) = \hat{Q}_{\tau_{t,\ell}(t)}(\tilde{\ell}) \), similar to the definition of \( Q_t^\ell \) given by Eq. \( 3.8 \). (By default \( \hat{Q}_t(0,0,0) = 0 \) for all \( \tau \).) The iteration \( 4.1 \) differs from the Q-learning iteration \( 3.8 \) in that instead of maximizing over \( \sigma \in V(s) \), we fix \( \sigma = \hat{\nu}_s \).

**Lemma 4.1.** If \( \{ \hat{Q}_t \} \) is bounded below w.p.1, so is \( \{ Q_t \} \).

**Proof.** We show by induction that \( Q_t \geq \hat{Q}_t \) for all \( t \). For \( t = 0 \), this holds since \( \hat{Q}_0 = Q_0 \) by definition. Suppose that for some \( t \geq 0 \), the desired relation holds for all \( \tau \leq t \). Then, for every \( \ell = (i, u, v) \in R \), using Eqs. \( 3.8 \), \( 4.1 \), the induction hypothesis, and the fact that \( \gamma_{t,\ell} \in [0,1] \), we have that

\[
Q_{t+1}(i, u, v) \geq (1 - \gamma_{t,\ell})Q_t(i, u, v) + \gamma_{t,\ell} \left( \hat{g}(i, u, v, s) + \inf_{\rho \in \tilde{U}(s)} \hat{Q}_t^\ell(s, \rho, \tilde{\nu}_s) \right) \\
\geq (1 - \gamma_{t,\ell})Q_t(i, u, v) + \gamma_{t,\ell} \left( \hat{g}(i, u, v, s) + \inf_{\rho \in \tilde{U}(s)} \hat{Q}_t^\ell(s, \rho, \tilde{\nu}_s) \right) = \hat{Q}_{t+1}(i, u, v),
\]

where \( s \) is a shorthand for the successor state \( j_t^\ell \). This completes the induction and establishes that \( Q_t \geq \hat{Q}_t \) for all \( t \). Hence \( \{ Q_t \} \) is bounded below w.p.1 if \( \{ \hat{Q}_t \} \) is so. \( \square \)

By the preceding lemma, in order to establish the lower boundedness of the Q-learning iterates \( \{ Q_t \} \), it is sufficient to prove that the sequence \( \{ \hat{Q}_t \} \) defined above is bounded below w.p.1. The iterates \( \{ Q_t \} \) are similar to Q-learning iterates in an MDP. Our goal now is to make this connection more precise so that we can apply the results or proof techniques developed for analyzing Q-learning in single-player problems to bound \( \{ Q_t \} \) from below. To this end, let us examine the single-player
problem faced by player I when player II plays the essentially proper policy \( \hat{\nu} \). We will call this single-player SSP problem \( \text{SSP}(\hat{\nu}) \). For later use, we will augment its state space to include the set \( R \) also. Here is the precise definition of \( \text{SSP}(\hat{\nu}) \).

**Definition 4.1.** \( \text{SSP}(\hat{\nu}) \) denotes the following single-player SSP problem:

1. The state space is \( S_0 \cup R \), with state 0 being a cost-free termination state.
2. From a state \( \ell = (i,u,v) \in R \), the system transitions to a state \( j \in S_0 \). The transition is uncontrolled and occurs with probability \( p_{\nu,ij}(u,v) \), and the expected one-stage cost is \( g(i,u,v) \).
3. For a state \( i \in S \), the control set is \( U(i) \), and for each \( u \in U(i) \), the system transitions to a state \( j \in S_0 \) with probability

\[
p_{\nu,ij}(u) = \sum_{v \in V(i)} \hat{\nu}(v \mid i) p_{ij}(u,v),
\]

and the expected one-stage cost is

\[
g_{\nu}(i,u) = \sum_{v \in V(i)} \hat{\nu}(v \mid i) g(i,u,v).
\]

Because \( \hat{\nu} \) is an essentially proper policy of the SSP game (cf. Definition 2.3), we can show that the single-player problem \( \text{SSP}(\hat{\nu}) \) satisfies the single-player SSP Model Assumption (cf. the discussion preceding Definition 2.3). Let \( \Pi_{\text{SD}} \) (\( \Pi_{\text{SR}} \)) denote the set of stationary deterministic (randomized) policies in \( \text{SSP}(\hat{\nu}) \).

**Lemma 4.2.** \( \text{SSP}(\hat{\nu}) \) satisfies the SSP Model Assumption; that is, there exists a proper policy in \( \Pi_{\text{SD}} \), and every improper policy in \( \Pi_{\text{SD}} \) incurs infinite cost for at least one initial state. (Here proper and improper policies are as defined in Definition 2.3 for a single-player SSP problem.)

**Proof.** The system dynamics of \( \text{SSP}(\hat{\nu}) \) described in Definition 2.3(2)-(3) shows that to prove the lemma, it suffices to consider only those states in \( S_0 \) and prove that \( \text{SSP}(\hat{\nu}) \) restricted to \( S_0 \) satisfies the SSP model Assumption. Let us simply call this restricted problem \( \text{SSP}(\hat{\nu}) \) in the proof below. Since \( \hat{\nu} \) is an essentially proper policy of a finite-space SSP game that satisfies Assumption 4.1 (by Definition 2.3 b), \( \text{SSP}(\hat{\nu}) \) has the following properties: there exists a proper policy in \( \Pi_{\text{SR}} \), and every improper policy in \( \Pi_{\text{SR}} \) incurs infinite cost for at least one initial state. Hence, to prove the lemma, we need to show that \( \text{SSP}(\hat{\nu}) \) has a proper policy in \( \Pi_{\text{SD}} \).

We claim that if there exists a proper policy in \( \Pi_{\text{SR}} \), then there must exist a proper policy in \( \Pi_{\text{SD}} \). This follows from the relation between the limiting average state-action frequency of a stationary randomized policy and the set of the limiting average state-action frequencies of all stationary deterministic policies, in a finite-space MDP. (For the definition of these limiting frequencies, see [Put94, Section 8.9.1].) In particular, consider any initial state distribution \( \alpha \) such that \( \alpha(i) > 0 \) for all \( i \in S \). Let \( \mu \in \Pi_{\text{SR}} \) be a proper policy. Let \( y_{\mu,\alpha} = \{ y_{\mu,\alpha}(i,u) \mid i \in S_0, u \in U(i) \} \) denote the limiting average state-action frequency of \( \mu \) for the initial state distribution \( \alpha \) (here the control set for the termination state is set to be \( U(0) = \{0\} \)). By [Put94, Theorem 8.9.3, p. 400], \( y_{\mu,\alpha} \) lies in the convex hull of the limiting average state-action frequencies of stationary deterministic policies for the initial distribution \( \alpha \). Since \( \mu \) is proper, the termination state 0 is reached w.p.1 for all initial states in \( S \), and consequently, \( y_{\mu,\alpha} \) is the vector with \( y_{\mu,\alpha}(0,0) = 1 \) for the termination state \( i = 0 \) and with \( y_{\mu,\alpha}(i,u) = 0 \) for \( (i,u) \neq (0,0) \). This vector must be an extreme point of the convex hull just mentioned (which is a subset of probability distributions on \( \{(i,u) \mid i \in S_0, u \in U(i)\} \)). Therefore, there exists some \( \mu_{\text{det}} \in \Pi_{\text{SD}} \) whose limiting average state-action frequency for the initial distribution \( \alpha \) equals 1 at \( (i,u) = (0,0) \). Since \( \alpha(i) > 0 \) for all \( i \in S \), this implies that the termination state 0 is reached w.p.1 for all initial states in \( S \) under the deterministic policy \( \mu_{\text{det}} \). Hence \( \mu_{\text{det}} \) is a proper policy in \( \Pi_{\text{SD}} \). This proves our claim. \( \square \)
For an SSP satisfying the SSP Model Assumption, the classical Q-learning algorithm generates a sequence of iterates that is bounded w.p.1, as proved by Yu and Bertsekas [YB13a]. The iterates \( \{\hat{Q}_t\} \) defined by Eq. (4.4) are similar to the classical Q-learning iterates, except for a small difference: in iteration (4.4), the minimization over the controls at the successor state is done after taking weighted averages of Q-factors (weighted according to \( \hat{\nu} \)), whereas there is no such averaging in classical Q-learning. This difference is mostly algebraic, however. Our subsequent proof of the lower boundedness of \( \{\hat{Q}_t\} \) follows essentially the lower boundedness proof given in [YB13a Section 3.3] for classical Q-learning.

4.1.2 Auxiliary sequence \( \{\tilde{Q}_t\} \)

We proceed to prove that \( \{\tilde{Q}_t\} \) given by iteration (4.4) is bounded below w.p.1, as proved by Yu and Bertsekas [YB13a]. The iterates \( \{\tilde{Q}_t\} \) are defined by Eq. (4.1) are similar to the classical Q-learning iterates, except for a small difference: for a weighted average of Q-factors, defined similarly to the notation \( \hat{Q}_t \) on the same probability space. The construction of this new process will be the key to our proof.

To this end, let us replace the inf operation in iteration (4.1) and write iteration (4.1) equivalently as follows. For every \( (i, u, v) \in R \) and \( t \geq 0 \),

\[
\tilde{Q}_{t+1}(i, u, v) = (1 - \gamma_{t, \ell})\tilde{Q}_t(i, u, v) + \gamma_{t, \ell}\left(\hat{g}(i, u, v, s) + \hat{Q}^{(\ell)}(s, u^\ell, \tilde{\nu})\right),
\]

where \( s \) is shorthand for the successor state \( j^\ell \), and \( u^\ell \) is a control such that 

\[
u^\ell \in \arg \min_{\hat{\nu} \in \hat{U}(s)} \tilde{Q}^{(\ell)}(s, \hat{\nu}),
\]

where the expression \( \tilde{Q}^{(\ell)}(s, \hat{\nu}) \) for \( \hat{\nu} \in \hat{U}(s) \) denotes the weighted average of the Q-factors:

\[
\tilde{Q}^{(\ell)}(s, \hat{\nu}) = \sum_{\tilde{\nu} \in \tilde{\nu}(s)} \tilde{\nu}(\tilde{\nu}) \tilde{Q}^{(\ell)}(s, \tilde{\nu}, \tilde{\nu}).
\]

Now consider an auxiliary sequence \( \{\tilde{Q}_t\} \) of the following form. Given some integer \( t_0 \) and Q-factor vector \( \hat{Q}_{t_0} \), let

\[
\tilde{Q}_t = \hat{Q}_{t_0}, \quad t \leq t_0,
\]

and let \( \tilde{Q}_{t+1}, t \geq t_0, \) be defined by the recursion: for every \( (i, u, v) \in R \),

\[
\tilde{Q}_{t+1}(i, u, v) = (1 - \gamma_{t, \ell})\tilde{Q}_t(i, u, v) + \gamma_{t, \ell}\left(\hat{g}(i, u, v, s) + \hat{Q}^{(\ell)}(s, u^\ell, \tilde{\nu})\right),
\]

where \( s \) is a shorthand for the successor state \( j^\ell \), and the expression \( \hat{Q}^{(\ell)}(s, u^\ell, \tilde{\nu}) \) is shorthand for a weighted average of Q-factors, defined similarly to the notation \( \hat{Q}^{(\ell)}(s, \hat{\nu}) \) above:

\[
\hat{Q}^{(\ell)}(s, u^\ell, \tilde{\nu}) = \sum_{\tilde{\nu} \in \tilde{\nu}(s)} \tilde{\nu}(\tilde{\nu}) \hat{Q}^{(\ell)}(s, u^\ell, \tilde{\nu}),
\]

with \( \hat{Q}^{(\ell)} \) representing the vector of Q-factors whose components are given by \( \hat{Q}^{(\ell)}(\tilde{\ell}) = \hat{Q}_{\tau_{\ell t}(\tilde{\ell})}(\tilde{\ell}), \tilde{\ell} \in R \cup \{0, 0, 0\} \) [by default \( \hat{Q}(0, 0, 0, 0) = \hat{Q}(0, 0, 0, 0, 0) = 0 \) for all \( \ell \)]. Most importantly, the variables \( \gamma_{t, \ell}, j^\ell, u^\ell, \) and \( \tau_{\ell t}(\ell) \) where \( \ell, \tilde{\ell} \in R, t \geq 0 \), in the definition (4.4) for \( \{\hat{Q}_t\} \) are the same random variables that appear in the iteration (4.4) that defines \( \{\hat{Q}_t\} \).

**Lemma 4.3.** Consider any sample path. Then for any values of \( t_0 \) and \( \hat{Q}_{t_0} \), \( \{\tilde{Q}_t\} \) is bounded below if and only if \( \{\hat{Q}_t\} \) given by Eqs. (4.4)-(4.6) is bounded below.
Proof. For every \( \ell = (i, u, v) \in R \) and \( t \geq t_0 \), using Eqs. (4.3), (4.6), and the fact that \( \gamma_{t, t} \in [0, 1] \), we have that
\[
| \tilde{Q}_{t+1}(i, u, v) - \tilde{Q}_t(i, u, v) | \leq (1 - \gamma_{t, t}) | \tilde{Q}_t(i, u, v) - \tilde{Q}_t(i, u, v) |
+ \gamma_{t, t} \sum_{v' \in V(s)} \tilde{u}_s(v') | \tilde{Q}_s^{(t)}(s, u', v') - \tilde{Q}_s^{(t)}(s, u', v') |
\leq \max_{\tau \leq \ell} \| \tilde{Q}_\tau - \tilde{Q}_\tau \|_\infty,
\]
where \( s = j_t' \). This implies that for all \( t \geq t_0 \),
\[
\max_{\tau \leq \ell} \| \tilde{Q}_\tau - \tilde{Q}_\tau \|_\infty \leq \max_{\tau \leq \ell} \| \tilde{Q}_\tau - \tilde{Q}_\tau \|_\infty.
\]
Hence, on a sample path, \( \{ \tilde{Q}_t \} \) is bounded below if and only if \( \{ \tilde{Q}_t \} \) is bounded below. \( \square \)

The sequence \( \{ \tilde{Q}_t \} \) is more convenient to work with than \( \{ \tilde{Q}_t \} \), because by Lemma 4.3 we have the freedom to choose for each sample path the initial time \( t_0 \) and initial value \( Q_{t_0} \) so that the resulting sequence \( \{ \tilde{Q}_t \} \) has a certain desirable structure. In the next step of the proof, we will make such a choice that will equate \( \{ \tilde{Q}_t \} \) to the costs in certain single-player SSP problems “neighboring” SSP(\( \bar{\nu} \)), in the sense that the parameters of these SSP problems lie close to those of SSP(\( \bar{\nu} \)).

Before we proceed, we need some notation and definitions for various neighborhoods of the model parameters, which we will use throughout the rest of the proof.

### 4.1.3 Some notation and definitions

As before, for a finite set \( A \), we denote by \( \mathcal{P}(A) \) the set of probability distributions on \( A \), and for \( a \in A \) and \( p \in \mathcal{P}(A) \), we write \( p(a) \) for the probability of \( a \) under \( p \). The support of \( p \), denoted \( \text{supp}(p) \), is the set \( \{ a \in A | p(a) \neq 0 \} \). For \( p_1, p_2 \in \mathcal{P}(A) \), we write \( p_1 \ll p_2 \) if \( p_1 \) is absolutely continuous with respect to \( p_2 \), that is, \( \text{supp}(p_1) \subset \text{supp}(p_2) \).

We use the following notation to represent the neighborhoods of the transition probability and one-stage cost parameters of SSP(\( \bar{\nu} \)) within certain affine subspaces. (The parameters of SSP(\( \bar{\nu} \)) are defined in Definition 4.1 and will be referred to below.)

- For each \( \ell = (i, u, v) \in R \), let \( p^\ell_o \in \mathcal{P}(S_o) \) denote the transition probability distribution at state \( \ell \) in SSP(\( \bar{\nu} \)), that is, \( p^\ell_o(j) = p_{ij}(u, v), \forall j \in S_o \). For each \( \delta > 0 \), define \( \mathcal{N}_\delta(p^\ell_o) \) to be the set of probability distributions that are not only in the \( \delta \)-neighborhood of \( p^\ell_o \) but also absolutely continuous with respect to \( p^\ell_o \), i.e.,
  \[
  \mathcal{N}_\delta(p^\ell_o) = \{ \textbf{d} \in \mathcal{P}(S_o) | |\textbf{d}(j) - p^\ell_o(j)| \leq \delta, \forall j \in S_o, \text{ and } \textbf{d} \ll p^\ell_o \}.
  \]

- Denote \( R^1_o = \{ (i, u) | i \in S_o, u \in U(i) \} \) where \( U(0) = \{ 0 \} \) denotes the (dummy) control set for the termination state 0. (This is the set of state-control pairs for player 1.) For each \( (i, u) \in R^1_o \), let \( p^{iu}_o \in \mathcal{P}(S_o) \) denote the transition probability distribution at state \( i \) with control \( u \) in SSP(\( \bar{\nu} \)): \( p^{iu}_o(j) = p_{ij, o}(u), \forall j \in S_o \). For each \( \delta > 0 \), define \( \mathcal{N}_\delta(p^{iu}_o) \) to be the subset of distributions in the \( \delta \)-neighborhood of \( p^{iu}_o \) that are absolutely continuous with respect to \( p^{iu}_o \):
  \[
  \mathcal{N}_\delta(p^{iu}_o) = \{ \textbf{d} \in \mathcal{P}(S_o) | |\textbf{d}(j) - p^{iu}_o(j)| \leq \delta, \forall j \in S_o, \text{ and } \textbf{d} \ll p^{iu}_o \}.
  \]

In particular, for \( (i, u) = (0, 0) \), \( p^{iu}_o(0) = 1 \) and \( \mathcal{N}_\delta(p^{iu}_o) = \{ p^{iu}_o \} \).

- Let \( q = \{ q(i, u, v) | (i, u, v) \in R \} \) be the vector of expected one-stage costs for the states in \( R \) in SSP(\( \bar{\nu} \)). Define \( \mathcal{N}_\delta(q) \) to be the \( \delta \)-neighborhood of \( q \) with \( c = \{ c(i, u, v) | (i, u, v) \in R \} \):
  \[
  \mathcal{N}_\delta(q) = \{ c | |c(i, u, v) - q(i, u, v)| \leq \delta, \forall (i, u, v) \in R \}.
  \]
• Similarly, let \( g_0 = \{ g_0(i, u) \mid (i, u) \in R^1_o \} \) be the vector of expected one-stage costs for the state-control pairs in \( R^1_o \) in SSP(\( \bar{v} \)). Define \( \bar{N}_\delta(g_0) \) to be the intersection of the \( \delta \)-neighborhood of \( g_0 \) with a subspace: with \( c = \{ c(i, u) \mid (i, u) \in R^1_o \} \),

\[
\bar{N}_\delta(g_0) = \{ c \mid |c(i, u) - g_0(i, u)| \leq \delta, \forall (i, u) \in R^1_o, \text{ and } c(0, 0) = 0 \}.
\]

For brevity, we will simply call the above sets \( \bar{N}_\delta(p^{o}_0), \bar{N}_\delta(p^{u}_0), \bar{N}_\delta(g) \) and \( \bar{N}_\delta(g_0) \) the \( \delta \)-neighborhoods of the respective parameters of SSP(\( \bar{v} \)).

### 4.1.4 Choose \( t_0 \) and initial \( \bar{Q}_{t_0} \) for a sample path and \( \delta > 0 \)

To initialize the auxiliary sequence \( \{ \bar{Q}_t \} \) defined by Eqs. (4.8), (4.9), we will choose time \( t_0 \) and vector \( \bar{Q}_{t_0} \) based on the information of an entire sample path. First, we define two random sequences on the same probability space as the process \( \{ \bar{Q}_t \} \): a sequence \( \{ \bar{g}_t \} \) of one-stage cost vectors, and a sequence \( \{ q_t \} \) of collections of probability distributions on \( S_0 \). They will be used to determine the values of \( t_0 \) and \( \bar{Q}_{t_0} \) on a sample path, for any chosen \( \delta > 0 \).

The sequence \( \{ \bar{g}_t \} \) can be related to the empirical one-stage costs and is defined recursively as follows. For \( t \geq 0 \),

\[
\bar{g}_{t+1}(i, u, v) = (1 - \gamma_{t, t}) \bar{g}_t(i, u, v) + \gamma_{t, t} \bar{g}(i, u, v, \bar{j}_t), \quad \forall (i, u, v) \in R,
\]

(4.7)

with \( \bar{g}_0(\cdot) \equiv 0 \) for \( t = 0 \). By the standard theory of stochastic approximation (see e.g., [BT96, Prop. 4.1 and Example 4.3, p. 141-143] or [KY03, Bor08]), Eqs. (4.8) and (4.11) imply that

\[
\bar{g}_t(i, u, v) \xrightarrow{a.s.} g(i, u, v), \quad \forall (i, u, v) \in R,
\]

(4.8)

where “a.s.” stands for “almost surely,” “w.p.1.”

The sequence \( \{ q_t \} \) can be related to empirical frequencies of state transitions and is defined recursively as follows. It has \( |R| \) component sequences, \( \{ q_{0\ell}^{\ell} \} , \ell \in R \). For each \( \ell = (i, u, v) \in R \), let

\[
q_{0\ell}^{\ell} \in \mathcal{P}(S_0), \quad q_{0\ell}^{\ell} \ll p_{0\ell}^{\ell},
\]

and let

\[
q_{\ell+1}^{\ell} = (1 - \gamma_{t, t}) q_{\ell}^{\ell} + \gamma_{t, t} e_j^\ell, \quad t \geq 0,
\]

(4.9)

where for \( j \in S_0 \), \( e_j \in \mathcal{P}(S_0) \) denotes the distribution with \( e_j(j) = 1 \). Then for all \( \ell = (i, u, v) \in R \), we have

\[
q_{\ell}^{\ell} \xrightarrow{a.s.} p_{\ell}^{\ell} \quad \text{and} \quad q_{\ell}^{\ell} \ll p_{\ell}^{\ell} \quad \text{w.p.1,} \quad \forall t \geq 0,
\]

(4.10)

where the first relation follows from Eqs. (3.9), (3.11) and the standard theory of stochastic approximation, and the second relation follows from the fact that \( j_t^\ell \) is a random successor state of state \( i \) with controls \( (u, v) \).

Equations (4.8), (4.10) indicate that the sequences \( \{ \bar{g}_t(i, u, v) \} \) and \( \{ q_{\ell}^{\ell} \} , \ell = (i, u, v) \in R \), converge to the corresponding one-stage cost and transition probability parameters of SSP(\( \bar{v} \)). We then obtain the following lemma, with which we will choose the initial time \( t_0 \).

**Lemma 4.4.** There exists a set of probability one on which, given any \( \delta > 0 \), there is a path-dependent time \( t_0 \) such that

\[
q_{\ell}^{\ell} \in \bar{N}_\delta(p_{\ell}^{\ell}), \quad \bar{g}_t \in \bar{N}_\delta(g), \quad \forall \ell \in R, \quad t \geq t_0.
\]

(4.11)

In the rest of the proof, we consider any sample path from the set of probability one given in Lemma 4.4. For any given \( \delta > 0 \), we choose \( t_0 \) given in Lemma 4.4 to be the initial time of the auxiliary sequence \( \{ \bar{Q}_t \} \). (Note that \( t_0 \) depends on the entire path and hence so does \( \bar{Q}_t \) for all \( t \).)
We now define the initial $\hat{Q}_{t_0}$. Let us fix some policy $\mu \in \Pi_{1,sk}$ that is proper for the single-player problem $SSP(\nu)$. (Such a policy exists because $\nu$ is an essentially proper policy of the game and $SSP(\nu)$ satisfies the SSP Model Assumption; cf. the proof of Lemma 4.2, Section 4.1.1.) Associate with $t_0$ and each $\ell = (i, u, v) \in R$ a time-inhomogeneous Markov chain $(i_0, u_0, v_0), (i_1, u_1), (i_2, u_2), \ldots$ with time-varying one-stage costs as follows:

- The initial state of this Markov chain is $(i_0, u_0, v_0) = (i, u, v)$. The state space for time $k = 0$ is $R$ and for time $k \geq 1$ is $S_0 \times U$ where $U = \cup_{s \in S_0} U(i)$ (the control space of player I).

- The probability distribution of this Markov chain, denoted $P_{t_0}^\ell$, is defined by time-varying transition probabilities: for all $(i, u, v) \in R_1^1$,

\[
P_{t_0}^\ell(i_1 = j, u_1 = w | i_0 = i, u_0 = u, v_0 = v) = q_{t_0}^\ell(j) \cdot \mu(w | j), \quad \text{for } k = 1, \tag{4.12}
\]

\[
P_{t_0}^\ell(i_k = j, u_k = w | i_{k-1} = i, u_{k-1} = u) = p_{t_0}^\ell(j) \cdot \mu(w | j), \quad \text{for } k \geq 2. \tag{4.13}
\]

The transition probabilities at those $(i, u, v) \not\in R_1^1$ can be defined arbitrarily because the chain has zero probability to visit such state-control pairs at any time, in view of the fact that $\mu$ is a policy for $SSP(\nu)$.

- Define time-varying one-stage cost functions $g_{t_0}^{\ell} : R \rightarrow R$ and $g_k^{\ell} : R_1^1 \rightarrow R$, $k \geq 1$, to be

\[
g_{t_0}^\ell = \tilde{g}_{t_0}, \quad g_k^\ell = g_{\nu}, \quad k \geq 1. \tag{4.14}
\]

For $k \geq 1$, we extend $g_k^{\ell}$ to $S_0 \times U$ by defining its values outside the domain $R_1^1$ to be $+\infty$, and we will treat $0 \cdot +\infty = 0$. This convention will be followed throughout.

We now define for every $\ell = (i, u, v) \in R$,

\[
\hat{Q}_{t_0}(i, u, v) = g_{t_0}^\ell(i, u, v) + E_P^{t_0}\left[ \sum_{k=1}^{\infty} g_k^{\ell}(i_k, u_k) \right], \tag{4.15}
\]

where $P_{t_0}^\ell$ in the superscript indicates that the expectation is taken with respect to it. The above expectation is well-defined and finite, and furthermore, the order of summation and expectation can be exchanged, i.e.,

\[
\hat{Q}_{t_0}(i, u, v) = g_{t_0}^\ell(i, u, v) + \sum_{k=1}^{\infty} E_P^{t_0}\left[ g_k^{\ell}(i_k, u_k) \right].
\]

This is because according to the preceding definition of the Markov chain associated with $t_0$ and $\ell = (i, u, v)$, under $P_{t_0}^\ell$, from time 1 onwards, the process $\{(i_k, u_k), k \geq 1\}$ evolves and incurs costs as in $SSP(\nu)$ under the proper policy $\mu$ [cf. Eqs. (4.12)-(4.14)], and consequently, $\sum_{k=1}^{\infty} |g_k^{\ell}(i_k, u_k)|$ is finite almost surely and its expectation is finite with respect to $P_{t_0}^\ell$.

The definition of $\hat{Q}_{t_0}$ above has two key properties:

(i) Each component $\hat{Q}_{t_0}(i, u, v)$ equals the expected total cost of some randomized Markov policy (which is $\mu$ here) in a time-inhomogeneous (single-player) SSP problem.

(ii) The parameters of that SSP problem, i.e., transition probabilities and one-stage costs, all lie in the $\delta$-neighborhoods $N_\delta(p_{t_0}^\ell), N_\delta(p_{t_0}^\nu), N_\delta(g), N_\delta(g_{t_0})$ of the corresponding parameters of $SSP(\nu)$.

We now show that these properties are preserved in $\hat{Q}_t$, $t \geq t_0$ defined by iteration [4.6].
4.1.5 Interpret \( \{\tilde{Q}_t\} \) as total costs in certain SSP problems neighboring SSP(\( \tilde{v} \))

The next lemma states precisely the interpretation we need of the auxiliary sequence \( \{\tilde{Q}_t\} \) resulting from the preceding choice of \( t_0 \) and \( \tilde{Q}_{t_0} \). Briefly speaking, each component of \( \tilde{Q}_t \), \( t \geq t_0 \), is equal to the expected total cost of a randomized Markov policy (represented by \( \{p_{k,t}^f, k \geq 1\} \) below) in a time-inhomogeneous SSP problem whose parameters (transition probabilities and one-stage costs, represented by \( \{p_{k,t}^f, g_k^f, k \geq 0\} \) below) lie in the \( \delta \)-neighborhoods of the corresponding parameters of SSP(\( \tilde{v} \)).

**Lemma 4.5.** Let the sequences \( \{\tilde{g}_t\} \) and \( \{\tilde{q}_t^f\} \), \( f = (i,u,v) \in R \) be as defined by Eqs. (4.7), (4.8), respectively. Consider any sample path from the set of probability one given in Lemma 4.4. For any \( \delta > 0 \), with \( t_0 \) and \( \tilde{Q}_{t_0} \) given as in Section 4.1.4 for the chosen \( \delta \), the sequence \( \{\tilde{Q}_t\} \) defined by Eqs. (4.7)-(4.8) has the following properties. For each \( f = (i,u,v) \in R \) and \( t \geq 0 \):

(a) \( \tilde{Q}_t(i,u,v) \) can be expressed as

\[
\tilde{Q}_t(i,u,v) = g_{0,t}^f(i_0,u_0,v_0) + \mathbb{E}^P_t\left[ \sum_{k=1}^{\infty} g_k^f(i_k,u_k) \right] = g_{0,t}^f(i_0,u_0,v_0) + \sum_{k=1}^{\infty} \mathbb{E}^P_t[g_k^f(i_k,u_k)]
\]

for some probability distribution \( P_{\ell,t}^f \) of a time-inhomogeneous Markov chain \((i_0,u_0,v_0),(i_1,u_1),(i_2,u_2),\ldots \) with \((i_0,u_0,v_0) = (i,u,v) \) and \((i_k,u_k) \in S_o \times U \) for \( k \geq 1 \), and for some one-stage cost functions \( g_{\ell,t}^f : R \rightarrow \mathbb{R} \), \( g_k^f : R^1_o \rightarrow \mathbb{R} \), \( k \geq 1 \) (with \( g_k^f \equiv +\infty \) on \( S_o \times U \setminus R^1_o \)).

(b) The transition probabilities of the Markov chain in (a) are time-varying and have the following product form: for all \((i,u),(\tilde{j},\tilde{w}) \in R^1_o\),

\[
P_{\ell,t}^f(i_1 = \tilde{j}, u_1 = \tilde{w} | i_0 = i, u_0 = u, v_0 = v) = p_{\ell,t}^f(\tilde{j} | i, u, v) \cdot \mu_{\ell,t}^f(\tilde{w} | \tilde{j}), \quad \text{for } k = 1,
\]

\[
P_{\ell,t}^f(i_k = \tilde{j}, u_k = \tilde{w} | i_{k-1} = \tilde{i}, u_{k-1} = \tilde{u}) = p_{\ell,t}^f(\tilde{j} | \tilde{i}, \tilde{u}) \cdot \mu_{\ell,t}^f(\tilde{w} | \tilde{j}), \quad \text{for } k \geq 2,
\]

where \( p_{\ell,t}^f \) and \( \mu_{\ell,t}^f \) are conditional probability distributions such that for all \( k \geq 1 \) and \((\tilde{i},\tilde{u}) \in R^1_o\), \( \tilde{j} \in S_o\),

\[
p_{\ell,t}^f(\cdot | \tilde{i}, \tilde{u}) \in \mathcal{N}_\delta(\mathcal{P}_\nu^{\tilde{i}u}), \quad \mu_{\ell,t}^f(\cdot | \tilde{j}) \in \mathcal{P}(U) \text{ with supp}(\mu_{\ell,t}^f(\cdot | \tilde{j})) \subset U(\tilde{j}),
\]

and for \( k = 0 \), \( p_{0,t}^f(\cdot | i, u, v) \in \mathcal{N}_\delta(\mathcal{P}_\nu) \) with \( p_{0,t}^f(\cdot | i, u, v) = q_{0,t}^f \) when \( t \geq t_0 \).

(c) The one-stage cost functions \( g_k^f \) in (a) satisfy

\[
g_{0,t}^f \in \mathcal{N}_\delta(g), \quad g_k^f \in \mathcal{N}_\delta(g_p), \quad k \geq 1,
\]

with \( g_{0,t}^f(i,u,v) = \tilde{g}_t(i,u,v) \) for \( t \geq t_0 \).

(d) For the Markov chain in (a), there exists an integer \( k_1 \geq 1 \) such that \( \{(i_k,u_k), k \geq k_1\} \) evolves and incurs costs as in SSP(\( \tilde{v} \)) under the proper policy \( \mu; \) i.e., for \( k \geq k_1 \),

\[
\mu_{k,t}^f(\cdot | \tilde{i}) = \mu(\cdot | \tilde{i}), \quad p_{k,t}^f(\cdot | \tilde{i}, \tilde{u}) = \mathcal{P}_\nu^{\tilde{i}u}, \quad g_k^f \tilde{i}, \tilde{u}) = g_p(\tilde{i}, \tilde{u}), \quad \forall (\tilde{i}, \tilde{u}) \in R^1_o.
\]

The proof of Lemma 4.5 is by induction on \( t \) for each \((i,u,v) \in R \). In the proof, we construct the one-stage cost functions and transition probabilities for the time-inhomogeneous Markov chain associated with \( t + 1 \), and this procedure resembles the construction of a cost-equivalent Markov policy in the classical MDP theory, for any given history-dependent policy and any given initial state. Other than the construction, the proof of Lemma 4.5 consists of mostly straightforward verifications of the properties (a)-(d) in the statement. Nevertheless, the verifications turn out to be lengthy, so we give the proof of Lemma 4.5 in 4.2.
4.1.6 Lower boundedness of \( \{ \tilde{Q}_t \} \)

We now come to the final step of our boundedness analysis: to lower-bound the optimal total costs of those time-inhomogeneous SSP problems neighboring SSP(\( \tilde{\nu} \)) and thereby lower-bound \( \{ \tilde{Q}_t \} \). As we have shown with Lemmas 4.4 and 4.5 for each sample path from a set of probability one, and for each \( \delta > 0 \), we can construct a sequence \( \{ \tilde{Q}_t \} \) such that \( \tilde{Q}_t(i, u, v) \) for each \( (i, u, v) \in R \) and \( t \geq 0 \) is the expected total cost of a randomized Markov policy in an SSP problem that has time-varying transition and one-stage cost parameters lying in the \( \delta \)-neighborhoods of the respective parameters of SSP(\( \tilde{\nu} \)). As we show below, when \( \delta \) is sufficiently small, the total costs in all such neighboring SSP problems can be bounded uniformly from below.

Let us be precise about the type of SSP problems involved here. Consider all time-inhomogenous SSP problems that have the same state-control space as SSP(\( \tilde{\nu} \)) and have initial states in \( R \) [cf. the definition of SSP(\( \tilde{\nu} \)) given in Definition 1.1]. For such an SSP, denote by \( p_k \) and \( g_k \) its state transition probability and its one-stage cost function, respectively, at the \( k \)-th stage. Let us call such an SSP a \( \delta \)-perturbed version of SSP(\( \tilde{\nu} \)) if for \( k = 0 \),

\[
g_0 \in \tilde{N}_\delta(g), \quad p_0(\cdot | i, u, v) \in \tilde{N}_\delta(p_\ell), \quad \forall \ell = (i, u, v) \in R,
\]

and for \( k \geq 1 \),

\[
g_k \in \tilde{N}_\delta(g), \quad p_k(\cdot | i, u) \in \tilde{N}_\delta(p_\ell), \quad \forall (i, u) \in R_i^k.
\]

Because SSP(\( \tilde{\nu} \)) satisfies the SSP Model Assumption (Lemma 1.2), we have the following lemma. It was proved in Yu and Bertsekas [YB13a, Section 3.3.4] and we will not repeat the proof here. The idea of the proof is to consider a time-homogeneous compact-control SSP problem where the controls include not only the regular controls but also the transition /one-stage cost parameters in the \( \delta \)-neighborhoods of the respective parameters of SSP(\( \tilde{\nu} \)), and to show that the optimal total costs of this compact-control SSP are finite when \( \delta \) is sufficiently small, by using a continuity argument together with the fact that SSP(\( \tilde{\nu} \)) satisfies the SSP model assumption and hence has finite optimal total costs by [BT91].

**Lemma 4.6 (YB13a Section 3.3.4).** There exist \( \delta > 0 \) and a finite constant \( C \) such that for all \( \delta \in [0, \delta] \), the optimal total cost of any \( \delta \)-perturbed version of SSP(\( \tilde{\nu} \)), for any initial state, is greater than \( C \).

Combining Lemma 4.5 with Lemma 4.6 we obtain the boundedness of \( \{ \tilde{Q}_t \} \) as stated below.

**Lemma 4.7.** Let \( \delta \in (0, \delta] \) where \( \delta \) is as given in Lemma 4.6. Then on any sample path from the set of probability one given in Lemma 4.4 with \( t_0 \) and \( Q_0 \) defined as in Section 4.1.4, for the chosen \( \delta \), the sequence \( \{ Q_t \} \) defined by Eqs. (4.3)-(4.4) is bounded below.

Lemma 4.7 together with Lemma 4.3 implies that the sequence \( \{ \tilde{Q}_t \} \) generated by the iteration (4.1) is bounded below w.p.1, which in turn implies, by Lemma 4.1 that the Q-learning iterates \( \{ Q_t \} \) are bounded below w.p.1. A symmetric argument then yields that \( \{ Q_t \} \) is bounded above w.p.1, as we explained at the beginning of Section 4.1. This completes the proof of Theorem 3.1 on the boundedness of Q-learning iterates \( \{ Q_t \} \) for SSP games satisfying Assumption 3.1.

### 4.2 Boundedness Analysis for a Special Case

If instead of Assumption 3.1, we make a stronger model assumption on the SSP game, then there is a simpler proof of the boundedness (and hence convergence) of Q-learning iterates, based on a contraction argument. We present this analysis to conclude Section 4.

First, let us define a notion of proper policies for an SSP game and use it to formulate the stronger model assumption just mentioned. (Our definition of properness differs slightly from that
in \([PB99]\). Recall that a pair of policies of the two players is non-prolonging if under those policies, the termination state is reached w.p.1 for all initial states.

**Definition 4.2** (Proper Policies in a Finite-Space SSP Game). For a finite-space SSP game, we say a policy \( \mu \in \Pi_{1,\text{SR}} \) of player I is proper if for every policy \( \nu \in \Pi_{2,\text{SR}} \) of player II, \((\mu, \nu)\) is non-prolonging. Similarly, we say a policy \( \nu \in \Pi_{2,\text{SR}} \) of player II is proper if for every policy \( \mu \in \Pi_{1,\text{SR}} \) of player I, \((\mu, \nu)\) is non-prolonging.

**Assumption 4.1.** The following holds in a finite-space SSP game:

(i) Every player has a proper stationary randomized policy.

(ii) For any pair of policies \((\mu, \nu) \in \Pi_{1,\text{SR}} \times \Pi_{2,\text{SR}}\) that is prolonging, \( J(i; \mu, \nu) = +\infty \) or \(-\infty \) for at least one initial state \(i\).

By Definition 4.2 if a player plays a proper policy, the optimal total costs or rewards for the other player are finite for all initial states. Therefore, an SSP game that satisfies the model conditions in Assumption 4.1 also satisfies Assumption 3.1. Consequently, the optimality results of Section 3 and the boundedness and convergence theorems for Q-learning hold under Assumption 4.1 as well. However, for proper policies of either players, the dynamic programming operators of their associated single-player problems exhibit a contraction property. This gives us a shortcut to prove the boundedness of Q-learning iterates under Assumption 4.1(i), without resorting to the long, general-case proof given earlier.

**Proposition 4.1** (Boundedness of Q-Learning Iterates in the Presence of Proper Policies). Suppose there exists a proper policy \( \bar{\nu} \in \Pi_{2,\text{SR}} \) of player I \((\bar{\nu} \in \Pi_{2,\text{SR}} \) of player II, respectively) in a finite-space SSP game. Then under Assumption 4.2(i)-(iii) and (v), for any given initial \(Q_0\), the sequence \(\{Q_t\}\) generated by the Q-learning algorithm (3.8) is bounded above (below, respectively) w.p.1.

We will prove the lower-boundedness part of Prop. 4.1. By symmetry, the upper-boundedness part of Prop. 4.1 follows from applying the same argument to the process \(-\{Q_t\}\). We start with a mapping \(F_{\bar{\nu}} : \mathbb{R}^{|R|} \rightarrow \mathbb{R}^{|R|}\) for a policy \(\bar{\nu} \in \Pi_{2,\text{SR}}\) defined by

\[
(F_{\bar{\nu}}Q)(i, u, v) := g(i, u, v) + \sum_{j \in S} p_{ij}(u, v) \inf_{\bar{u} \in U(j)} Q(j, \bar{u}, \bar{v}_j), \quad \forall \,(i, u, v) \in R, \forall \, Q,
\]

where \(\bar{v}_j = \bar{\nu}(\cdot | j)\) and

\[
Q(j, \bar{u}, \bar{v}_j) = \sum_{\bar{v} \in V(j)} \bar{v}_j(\bar{v}) Q(j, \bar{u}, \bar{v}).
\]

Given a positive vector \(\xi = \{\xi(i, u, v) | (i, u, v) \in R\}\), let \(\|\cdot\|_\xi\) denote the weighted sup-norm on the space of Q-factors given by \(\|Q\|_\xi = \max_{(i,u,v) \in R} \frac{|Q(i,u,v)|}{\xi(i,u,v)}\).

**Lemma 4.8.** For a proper policy \(\bar{\nu} \in \Pi_{2,\text{SR}}, F_{\bar{\nu}}\) given by Eq. (4.16) is a contraction with respect to some weighted sup-norm \(\|\cdot\|_\xi\), i.e., for some \(\beta \in [0, 1)\),

\[
\|F_{\bar{\nu}}Q - F_{\bar{\nu}}Q'\|_\xi \leq \beta \|Q - Q'\|_\xi, \quad \forall \, Q, Q'.
\]

**Proof.** First, we define a single-player SSP problem and use its optimal total costs to construct the weight vector \(\xi\) in the desired norm \(\|\cdot\|_\xi\), similar to the proofs of [BT96] Prop. 2.2, p. 23-24 and [PB99] Lemma 4.1. Consider a single-player SSP problem on the state space \(S_o \cup R\) where the system dynamics are the same as those of SSP(\(\bar{\nu}\)) given in Definition 4.1 and all the one-stage costs are \(-1\) except for that at the cost-free termination state \(0\). Because \(\bar{\nu}\) is a proper policy of player II, by the definition of a proper policy in an SSP game (cf. Definition 4.2), the single-player SSP
problem just defined satisfies the SSP Model Assumption (cf. Section 2.2). Therefore, by [BT91], its optimal total cost function \( J^* \) is finite at all states in \( S_o \cup R \) and satisfies the dynamic programming equation
\[
J^*(i, u, v) = -1 + \sum_{j \in S} p_{ij}(u, v) J^*(j), \quad \forall (i, u, v) \in R, \tag{4.17}
\]
\[
J^*(i) = -1 + \min_{u \in U(i)} \sum_{j \in S} p_{ij}(u) J^*(j), \quad \forall i \in S. \tag{4.18}
\]

We also have that \( J^*(i) \leq -1 \) and \( J^*(i, u, v) \leq -1 \) for every state \( i \) and \( (i, u, v) \), since the one-stage costs before termination are \(-1\).

Let us define
\[
\xi(i, u, v) = -J^*(i, u, v) \geq 1, \quad (i, u, v) \in R,
\]
\[
\xi_u(i, u) = \sum_{v \in V(i)} \bar{v}_i(v) \xi(i, u, v), \quad i \in S, \ u \in U(i).
\]
For every \( i \in S \) and \( u \in U(i) \), by Eqs. (4.17)-(4.18) and the definition of \( p_{v,ij} \) [cf. Definition 4.12],
\[
\xi_u(i, u) = -\left(-1 + \sum_{j \in S} p_{v,ij}(u) J^*(j)\right) \\
\leq -\left(-1 + \min_{u \in U(i)} \sum_{j \in S} p_{v,ij}(u) J^*(j)\right) = -J^*(i),
\]
and hence
\[
\sup_{u \in U(i)} \xi_u(i, u) \leq -J^*(i), \quad \forall i \in S.
\]

Then with \( \beta = \max_{(i,u,v) \in R} \frac{\xi(i,u,v)-1}{\xi(i,u,v)} \in [0, 1) \), we have for every \( (i, u, v) \in R \),
\[
\sum_{j \in S} p_{ij}(u, v) \sup_{\bar{u} \in U(J)} \xi_u(j, \bar{u}) \leq \sum_{j \in S} p_{ij}(u, v) (-J^*(j)) = \xi(i, u, v) - 1 \leq \beta \xi(i, u, v), \tag{4.19}
\]
where the equality follows from Eq. (4.17).

We now prove that \( F_{\bar{p}} \) is a contraction with respect to \( \| \cdot \|_\xi \) and with modulus \( \beta \). By the definition of \( F_{\bar{p}} \) [cf. Eq. (4.10)], for every \( (i, u, v) \in R \),
\[
\left| (F_{\bar{p}}Q)(i, u, v) - (F_{\bar{p}}Q')(i, u, v) \right| = \left| \sum_{j \in S} p_{ij}(u, v) \left( \inf_{\bar{u} \in U(J)} Q_u(j, \bar{u}) - \inf_{\bar{u} \in U(J)} Q_{\bar{u}}(j, \bar{u}) \right) \right|,
\]
where we define \( Q_u(j, \bar{u}) = \sum_{\tilde{v} \in V(j)} \bar{v}_j(\tilde{v}) Q(j, \tilde{v}, \bar{u}) \) and we define \( Q_{\bar{u}}(j, \bar{u}) \) similarly. Let \( \Delta = |Q - Q'| \) (the absolute values are taken component-wise). Using the preceding equation, we have for every \( (i, u, v) \in R \),
\[
\left| (F_{\bar{p}}Q)(i, u, v) - (F_{\bar{p}}Q')(i, u, v) \right| \leq \sum_{j \in S} p_{ij}(u, v) \sup_{\bar{u} \in U(J)} \sum_{\tilde{v} \in V(j)} \bar{v}_j(\tilde{v}) \Delta(j, \tilde{v}, \bar{u}) \\
= \sum_{j \in S} p_{ij}(u, v) \sup_{\bar{u} \in U(J)} \sum_{\tilde{v} \in V(j)} \bar{v}_j(\tilde{v}) \xi_u(j, \tilde{v}, \bar{u}) \frac{\Delta(j, \tilde{v}, \bar{u})}{\xi(j, \tilde{v}, \bar{u})} \\
\leq \| Q - Q' \|_{\xi} \sum_{j \in S} p_{ij}(u, v) \sup_{\bar{u} \in U(J)} \xi_u(j,\bar{u}) \\
\leq \| Q - Q' \|_{\xi} \beta \xi(i, u, v),
\]
where the last inequality follows from Eq. (4.19). This implies \( \| F_{\bar{p}}Q - F_{\bar{p}}Q' \|_{\xi} \leq \beta \| Q - Q' \|_{\xi} \). \hfill \Box
Proof of Prop. 4.1. We prove the lower-boundedness part of the proposition; as mentioned earlier, the upper-boundedness part follows from applying the same argument to the process \( \{-Q_t\} \).

Consider the process \( \{\hat{Q}_t\} \) defined by the iteration (4.1) with \( \bar{\nu} \) being a proper policy of player II. By Lemma 4.1, to prove that the sequence \( \{Q_t\} \) of Q-learning iterates is bounded below w.p.1, it is sufficient to prove that \( \{\hat{Q}_t\} \) is bounded below w.p.1. Now the iteration (4.1) for \( \{\hat{Q}_t\} \) can be equivalently written as: for every \( \ell = (i, u, v) \in R \) and \( t \geq 0 \),

\[
\hat{Q}_{t+1}(i, u, v) = (1 - \gamma_{t,\ell})\hat{Q}_t(i, u, v) + \gamma_{t,\ell}(F_{\bar{\nu}}\hat{Q}_t^{(\ell)})(i, u, v) + \gamma_{t,\ell} w_{t,\ell},
\]

where \( F_{\bar{\nu}} \) is the mapping given by (4.16) and \( w_{t,\ell} \) is a noise term given by

\[
w_{t,\ell} = \hat{g}(i, u, v, s) + \inf_{\rho \in U(s)} \hat{Q}_t^{(\ell)}(s, \rho, \bar{\nu}_s) - (F_{\bar{\nu}}\hat{Q}_t^{(\ell)})(i, u, v)
\]

with \( s = j^{\prime}_t \).

By Lemma 4.8, \( F_{\bar{\nu}} \) is a contraction with respect to a weighted sup-norm, so we can apply the result of [Tsi94] for asynchronous stochastic approximation algorithms involving contraction mappings. Direct calculation shows that for any given \( \hat{Q}_0 \), under Assumption 3.2(i)-(iii), the noise terms, \( w_{t,\ell}, \ell \in R, t \geq 0 \), satisfy the conditional mean and variance conditions required in the analysis of [Tsi94]: \( \mathbb{E}[w_{t,\ell} | F_t] = 0 \) w.p.1, and \( \mathbb{E}[w_{t,\ell}^2 | F_t] \leq A + B \max_{\ell \in R} \max_{t \leq T} |\hat{Q}_t^{(\ell)}|^2 \) w.p.1, for some deterministic constants \( A \) and \( B \). Therefore, by [Tsi94] Theorem 1, for any given \( \hat{Q}_0 = Q_0 \), \( \{\hat{Q}_t\} \) is bounded w.p.1 under Assumption 3.2(ii)-(iii) and (v). [Assumption 3.2(iv) is not needed for bounding the iterates, although it is needed for establishing their convergence.] As mentioned earlier, by Lemma 4.1 this implies that for any given initial \( Q_0 \), \( \{Q_t\} \) is bounded below w.p.1.

Acknowledgements

I thank Prof. Dimitri Bertsekas and Prof. John Tsitsiklis for helpful comments. This research was supported by the Air Force Grant FA9550-10-1-0412.

References

[ABB02] J. Abounadi, D. P. Bertsekas, and V. S. Borkar, Stochastic approximation for non-expansive maps: Application to Q-learning algorithms, SIAM J. on Control and Optimization 41 (2002), 1–22.

[BF68] D. Blackwell and T. S. Ferguson, The big match, Ann. Math. Stat. 39 (1968), 159–163.

[BK76] T. Bewley and E. Kohlberg, The asymptotic theory of stochastic games, Math. Oper. Res. 1 (1976), 197–208.

[BM00] V. S. Borkar and S. Meyn, The O.D.E. method for convergence of stochastic approximation and reinforcement learning, SIAM J. on Control and Optimization 38 (2000), 447–469.

[Bor98] V. S. Borkar, Asynchronous stochastic approximation, SIAM J. on Control and Optimization 36 (1998), 840–851.

[Bor08] ———, Stochastic approximation: A dynamic viewpoint, Hindustan Book Agency, New Delhi, 2008.

[BT91] D. P. Bertsekas and J. N. Tsitsiklis, An analysis of stochastic shortest path problems, Math. Oper. Res. 16 (1991), 580–595.

[BT96] ———, Neuro-dynamic programming, Athena Scientific, Belmont, MA, 1996.

[Eve57] H. Everett, Recursive games, Contributions to the Theory of Games (M. Dresher, A. W. Tucker, and P. Wolfe, eds.), Annals of Mathematics Studies 39, vol. III, Princeton University Press, Princeton, 1957, pp. 47–78.

[Fan53] K. Fan, Minimax theorems, Proc. Nat. Acad. Sci. U.S.A. 39 (1953), 42–47.
A. Federgruen, *On the functional equations in undiscounted and sensitive discounted stochastic games*, Z. Oper. Res. 24 (1980), 243–262.

E. A. Feinberg, *Stationary strategies in Borel dynamic programming*, Math. Oper. Res. 17 (1992), 392–397.

J. Filar and K. Vrieze, *Competitive Markov decision processes*, Springer-Verlag, New York, 1997.

D. Gillette, *Stochastic games with zero-stop probabilities*, Contributions to the Theory of Games (M. Dresher, A. W. Tucker, and P. Wolfe, eds.), Annals of Mathematics Studies 39, vol. III, Princeton University Press, Princeton, 1957, pp. 179–188.

H. J. Kushner and S. G. Chamberlain, *Finite state stochastic games: Existence theorems and computational procedures*, IEEE Trans. on Automatic Control AC-14 (1969), no. 3.

P. R. Kumar and T. H. Shiau, *Zero sum dynamic games*, Control and Dynamic Games (C. T. Leondes, ed.), Academic Press, N.Y., 1981, pp. 1345–1378.

H. J. Kushner and G. G. Yin, *Stochastic approximation and recursive algorithms and applications*, 2nd ed., Springer-Verlag, New York, 2003.

M. L. Littman, *Algorithms for sequential decision making*, Ph.D. thesis, Brown University, Providence, R.I., 1996.

J. F. Mertens and A. Neyman, *Stochastic games*, Int. J. Game Theory 10 (1981), 53–66.

A. S. Nowak, *Universally measurable strategies in zero-sum stochastic games*, Ann. Prob. 13 (1985), 269–287.

J. F. Mertens and A. Neyman, *Stochastic games*, Int. J. Game Theory 10 (1981), 53–66.

S. D. Patek and D. P. Bertsekas, *Stochastic shortest path games*, SIAM J. on Control and Optimization 37 (1999), 804–824.

M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*, John Wiley & Sons, New York, 1994.

L. S. Shapley, *Stochastic games*, Proceedings of the National Academy of Sciences U.S.A. 39 (1953), 1095–1100.

J. N. Tsitsiklis, *Asynchronous stochastic approximation and Q-learning*, Machine Learning 16 (1994), 185–202.

F. Thuijsman and O. J. Vrieze, *The bad match, a total reward stochastic game*, Oper. Res. Spektrum 9 (1987), 93–99.

F. Thuijsman and O. J. Vrieze, *The bad match, a total reward stochastic game*, Oper. Res. Spektrum 9 (1987), 93–99.

M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*, John Wiley & Sons, New York, 1994.

C. J. C. H. Watkins, *Learning from delayed rewards*, Ph.D. thesis, Cambridge University, England, 1989.

H. Yu and D. P. Bertsekas, *On boundedness of Q-learning iterates for stochastic shortest path problems*, Math. Oper. Res. 38 (2013), 209–227.

H. Yu and D. P. Bertsekas, *On boundedness of Q-learning iterates for stochastic shortest path problems*, Math. Oper. Res. 38 (2013), 209–227.

H. Yu and D. P. Bertsekas, *On boundedness of Q-learning iterates for stochastic shortest path problems*, Math. Oper. Res. 38 (2013), 209–227.
Appendix A  Proof of Lemma 4.5

The proof is by induction on $t$. For $t = t_0$, $Q_{t_0}$ satisfies the properties (a)-(d) in the lemma by its definition and our choice of the sample path and $t_0$ (cf. Lemma 4.4 and Eqs. (4.12)-(4.15)). Since $Q_t = Q_{t_0}$ for $t < t_0$, they also satisfy properties (a)-(d). So consider $t \geq t_0$ and suppose these properties are satisfied by all $Q_{\tau}$, $0 \leq \tau \leq t$. Let us show that they are satisfied by $Q_{t+1}$.

Consider $\tilde{Q}_{t+1}(i, u, v)$ for each $\ell = (i, u, v) \in R$. To simplify notation, denote $\gamma = \gamma_{t, \ell} \in [0, 1]$ (cf. Lemma 4.4). By Eq. (4.6),

$$\tilde{Q}_{t+1}(i, u, v) = (1 - \gamma) \tilde{Q}_t(i, u, v) + \gamma \left( \tilde{g}(i, u, v, s) + \mathcal{N}_t^{(s)}(s, u, s) \right),$$

where $s = j^t_\ell, \tilde{u} = u^t_\ell, \mathcal{Q}_t^{(s)}(s, \tilde{u}, \tilde{v}) = \sum_{\tilde{v} \in V(s)} \tilde{v}_s(\tilde{v}) \tilde{Q}_t^{(s)}(s, \tilde{u}, \tilde{v})$, and

$$\tilde{Q}_t^{(s)}(s, \tilde{u}, \tilde{v}) = \tilde{Q}_{\tau_t}(t)(s, \tilde{u}, \tilde{v}) \quad \text{with} \quad \tau_t = (s, \tilde{u}, \tilde{v}), \quad \tau_t(t) \leq t, \quad \forall \tilde{v} \in V(s).$$

Let us use the simplified notation $\tilde{\tau} = \tau_{t+1}(t)$ for $\tilde{v} \in V(s)$. By the induction hypothesis, we can express $\tilde{Q}_t(i, u, v)$ and each term $\tilde{Q}_{\tau_t}(t)(s, \tilde{u}, \tilde{v}), \tilde{v} \in V(s)$, for $s \neq 0$, in the form given in the statement (a) of the lemma. Thus when $s \neq 0$, we can write Eq. (A.1) as

$$\tilde{Q}_{t+1}(i, u, v) = (1 - \gamma) g_0^{\ell, t}(i, u, v) + (1 - \gamma) \sum_{k=1}^{\infty} \mathbb{E}^{P_t}_k \left[ g_k^{\ell, t}(i_k, u_k) \right]$$

$$+ \gamma \tilde{y}(i, u, v, s) + \gamma \sum_{\tilde{v} \in V(s)} \tilde{v}_s(\tilde{v}) \cdot \left( g_0^{\ell, \tau_\tilde{v}}(s, \tilde{u}, \tilde{v}) + \sum_{k=1}^{\infty} \mathbb{E}^{P_k}_t \left[ g_k^{\ell, \tau_\tilde{v}}(i_k, u_k) \right] \right)$$

$$= \sum_{k=0}^{\infty} C_k$$

where

$$C_0 = (1 - \gamma) g_0^{\ell, t}(i, u, v) + \gamma \tilde{y}(i, u, v, s),$$

$$C_1 = (1 - \gamma) \mathbb{E}^{P_t}_1 \left[ g_1^{\ell, t}(i_1, u_1) \right] + \gamma \sum_{\tilde{v} \in V(s)} \tilde{v}_s(\tilde{v}) \cdot g_0^{\ell, \tau_\tilde{v}}(s, \tilde{u}, \tilde{v}),$$

and

$$C_k = (1 - \gamma) \mathbb{E}^{P_t}_k \left[ g_k^{\ell, t}(i_k, u_k) \right] + \gamma \sum_{\tilde{v} \in V(s)} \tilde{v}_s(\tilde{v}) \cdot \mathbb{E}^{P_k}_t \left[ g_{k-1}^{\ell, \tau_\tilde{v}}(i_{k-1}, u_{k-1}) \right], \quad k \geq 2.$$

For the sake of convenience, let us define a few terms for the case $s = 0$ and make the above formulas valid for $s = 0$ as well. Recall that for $s = 0$, we have $U(0) = V(0) = \{0\}$ and with $\tilde{v} = 0$, $\ell_\tilde{v} = (0, 0, 0)$ and $\tilde{Q}_t(s, \tilde{u}, \tilde{v}) = \tilde{Q}_t(0, 0, 0) = 0$ for all $t$. Let us set $\tau_0 = 0$ in this case (since this term can be defined arbitrarily).

To express 0 in the form given in the lemma, let us simply define $P_t^{\ell, k}$ in this case to be the probability distribution of the Markov chain $\{(i_k, u_k), k \geq 0\}$ that starts from the absorbing termination state $(i_0, u_0) = (0, 0)$; let $g_0^{\ell, \tau_0}(0, 0, 0) = 0$ and let $g_0^{\ell, \tau_0}(s, \tilde{u}, \tilde{v})$ coincide with $g$ elsewhere; and let $g_k^{\ell, \tau_0} = g_0$, $k \geq 1$. With these definitions, we have $0 = g_0^{\ell, \tau_0}(\tilde{v}, \tilde{u}, \tilde{v}) + \sum_{k=1}^{\infty} \mathbb{E}^{P_k}_t \left[ g_k^{\ell, \tau_0}(i_k, u_k) \right]$ and Eq. (A.2) holds for $s = 0$. For later use, let us also define transition probabilities and other quantities so that some properties in the statement of the lemma hold for $s = 0$. In particular, let $\mu_k^{\ell, \tau_0} = \mu, \nu_k^{\ell, \tau_0} = \nu$ for $k \geq 1$ and $i_0, \tilde{u}_0 \in R^0_{\tilde{v}}$, and also let $P_0^{\ell, \tau_0} = P_0^0$ (i.e., $P_0^{\ell, \tau_0}(0, 0, 0) = 1$). Then $P_t^{\ell, k}$ can be expressed in the product form given in property (b), and it satisfies property (d) with $k_{\tau_0} = 1$. 33
We now rewrite each term $C_k$ in the above expression of $\tilde{Q}_{t+1}(i, u, v)$ in a desirable form, first for $k = 0$, then for $k \geq 2$, and finally, for $k = 1$. During this procedure, we will define the transition probabilities $p^k_{t+1}$ and $\mu^k_{t+1}$ that compose the probability distribution $P^k_{t+1}$ of the time-inhomogenous Markov chain for $t + 1$, as well as the one-stage cost functions $g^k_{t+1}$ required in the statement of the lemma.

**For $k = 0$:** By property (c) of the induction hypothesis, $g^0_{t,t}(i, u, v) = \tilde{g}_t(i, u, v)$. Using this and the definition of $\{\tilde{g}_t\}$ [cf. Eq. (4.7)], we have that

$$C_0 = (1 - \gamma) \tilde{g}_t(i, u, v) + \gamma \tilde{g}_t(i, u, v, s) = \tilde{g}_{t+1}(i, u, v). \quad (A.6)$$

Let the cost function $g^0_{t,t+1}$ and transition probability $p^0_{t+1}(\cdot | i, u, v)$ be

$$g^0_{t,t+1} = \tilde{g}_{t+1}, \quad p^0_{t+1}(\cdot | i, u, v) = q^0_{t+1}. \quad (A.7)$$

By Lemma 4.4 and our choice of the sample path, $g^0_{t,t+1}$ and $p^0_{t+1}$ satisfy the requirements in properties (b) and (c), that is,

$$g^0_{t,t+1} \in \mathcal{N}_d(g), \quad p^0_{t+1}(\cdot | i, u, v) \in \mathcal{N}_d(p^0).$$

**For $k \geq 2$:** Let $P^k_1$ denote the law of $(i_k, u_k, i_{k+1})$ under $P^k_t$, and for each $\tilde{v} \in V(s)$, let $P^k_{v,\tilde{v}}$ denote the law of $(i_{k-1}, u_{k-1}, i_k)$ under $P^k_{\tau_{\tilde{v}}}$. Let $P^k_3$ denote the convex combination of them,

$$P^k_3 = (1 - \gamma)P^k_1 + \gamma \sum_{\tilde{v} \in V(s)} \tilde{\nu}_s(\tilde{v}) \cdot P^k_{2,\tilde{v}}. \quad (A.8)$$

We regard these laws as probability measures on the sample space $\tilde{\Omega} = S_n \times U \times S_n$, and we denote by $X, Y$ and $Z$ the function that maps a point $\omega = (i, \tilde{u}, \tilde{j}) \in \tilde{\Omega}$ to its 1st, 2nd and 3rd coordinate, respectively. Using property (b) of $P^k_t$ and $P^k_{\tau_{\tilde{v}}}$ from the induction hypothesis (in particular, using the property of $\{\mu^k_{t}, k \geq 1\}$, $\{\mu^k_{t,\tau_{\tilde{v}}}, k \geq 1\}$), it is clear that $\text{supp}(P^k_3) \subset R^1_n \times S_n$, a subset of $\tilde{\Omega}$. So we can write the term $C_k$ in Eq. (A.3) for each $k \geq 2$ as

$$\sum_{i} \sum_{u} \left( (1 - \gamma) P^k_1(X = i, Y = u) \cdot g^k(i, u) + \gamma \sum_{\tilde{v} \in V(s)} \tilde{\nu}_s(\tilde{v}) \cdot P^k_{2,\tilde{v}}(X = i, Y = u) \cdot g^k_{t+1}(\tilde{i}, \tilde{u}) \right).$$

Next we will define the $k$th-stage cost function $g^k_{t,t+1}$ so that we can rewrite the above expression of $C_k$ equivalently as

$$C_k = \sum_{i} \sum_{u} \sum_{\tilde{v} \in U(i)} P^k_3(X = i, Y = u) \cdot g^k_{t,t+1}(i, u). \quad (A.9)$$

We will also define the transition probabilities $\mu^k_{t,t+1}(\cdot | i)$ and $p^k_{t,t+1}(\cdot | i, u)$ for all $(i, u) \in R^1_n$ so that we have for every $(\tilde{i}, \tilde{u}) \in R^1_n$ and $\tilde{j} \in S_n$,

$$P^k_3(X = i, Y = u, Z = j) = P^k_3(X = i, Y = u) \cdot P^k_{t+1}(\tilde{j} | i, u). \quad (A.10)$$

We define the cost function $g^k_{t,t+1}$ as follows. For each $(\tilde{i}, \tilde{u}) \in R^1_n$, if $P^k_3(X = \tilde{i}, Y = \tilde{u}) = 0$, let $g^k_{t,t+1}(\tilde{i}, \tilde{u}) = g_t(\tilde{i}, \tilde{u})$; otherwise, let

$$g^k_{t,t+1}(\tilde{i}, \tilde{u}) = \frac{(1 - \gamma) P^k_1(X = \tilde{i}, Y = \tilde{u})}{P^k_3(X = \tilde{i}, Y = \tilde{u})} \cdot g^k X = \tilde{i}, Y = \tilde{u}) + \gamma \sum_{\tilde{v} \in V(s)} \tilde{\nu}_s(\tilde{v}) \frac{P^k_{2,\tilde{v}}(X = \tilde{i}, Y = \tilde{u})}{P^k_3(X = \tilde{i}, Y = \tilde{u})} \cdot g^k_{t+1}(\tilde{i}, \tilde{u}). \quad (A.12)$$
Then, by the definition of $P^k_3$ [cf. Eq. (A.8)], Eq. (A.9) clearly holds. Observe from Eq. (A.12) that $g^{k,t+1}_k(i, \bar{u})$ is a convex combination of $g^{k,t}_k(i, \bar{u})$ and $g^{k,t-\tau_0}_k(i, \bar{u}), \bar{v} \in V(s)$. The latter terms, by property (c) of the induction hypothesis and by the definitions we gave for the case $s = 0$, all lie in the $\delta$-neighborhood of $g_v(i, \bar{u})$, and they all equal $g_v(i, \bar{u}) = 0$ if $(i, \bar{u}) = (0, 0)$. Hence, when $P^k_3(X = i, Y = \bar{u}) > 0$ and $g^{k,t+1}_k(i, \bar{u})$ is given by Eq. (A.12), $|g^{k,t+1}_k(i, \bar{u}) - g_v(i, \bar{u})| \leq \delta$, and $g^{k,t+1}_k(i, \bar{u}) = 0$ if $(i, \bar{u}) = (0, 0)$. This shows that $g^{k,t+1}_k$ satisfies the requirement in property (c) for $t + 1$: $g^{k,t+1}_k \in \tilde{N}_\delta(g_v)$.

Reasoning similarly, since by property (d) of the induction hypothesis, when $k \geq k_t$ and $k \geq \max_{\bar{v} \in V(s)} k_{\tau_0} + 1$, $g^{k,t}_k(i, \bar{u}) = g^{k,t-\tau_0}_k(i, \bar{u}) = g_v(i, \bar{u})$ for all $\bar{v} \in V(s)$ and $(i, \bar{u}) \in R^1_{\bar{u}}$, it follows that $g^{k,t+1}_k$ satisfies the requirement in property (d) for $t + 1$:

$$g^{k,t+1}_k = g_v, \quad \forall k \geq k_{t+1} := \max \{k_t, \max_{\bar{v} \in V(s)} k_{\tau_0} + 1\}.$$ 

Define the transition probability distributions $\mu^{k,t+1}_k$ and $p^{k,t+1}_k$ by

$$\mu^{k,t+1}_k(\cdot | \bar{i}) = P^k_3(Y = \cdot | X = \bar{i}), \quad \forall \bar{i} \in S_o,$$ 

$$p^{k,t+1}_k(\cdot | \bar{i}, \bar{u}) = P^k_3(Z = \cdot | X = \bar{i}, Y = \bar{u}), \quad \forall (\bar{i}, \bar{u}) \in R^1_{\bar{u}}.$$ 

In the right-hand sides of Eqs. (A.13)-(A.14), in case an event that is conditioned on has probability zero, the corresponding conditional probability, which can be defined arbitrarily, is defined according to the proper policy $\mu$ or the transition probabilities of SSP$(\bar{v})$ as:

$$P^k_3(Y = \cdot | X = \bar{i}) = \mu(\cdot | \bar{i}), \quad \text{if } P^k_3(X = \bar{i}) = 0;$$

$$P^k_3(Z = \cdot | X = \bar{i}, Y = \bar{u}) = p^\bar{v}{\bar{u}}, \quad \text{if } P^k_3(X = \bar{i}, Y = \bar{u}) = 0.$$ 

The desired equalities (A.10)-(A.11) then hold by these definitions. We now verify that $\mu^{k,t+1}_k$ and $p^{k,t+1}_k$ satisfy the requirements in properties (b) and (d) for $t + 1$.

First, we show that $p^{k,t+1}_k$ satisfies the requirement in property (b):

$$p^{k,t+1}_k(\cdot | \bar{i}, \bar{u}) \in \tilde{N}_\delta(p^{\bar{v}}{\bar{u}}), \quad \forall (\bar{i}, \bar{u}) \in R^1_{\bar{u}}.$$ 

This holds by definition if $P^k_3(X = \bar{i}, Y = \bar{u}) = 0$, so we consider the case $P^k_3(X = \bar{i}, Y = \bar{u}) > 0$. By the induction hypothesis and by the definitions we made for the case $s = 0$, $P^t_1$ and $P^{k,t}_2, \bar{v} \in V(s)$, all have the product form given in property (b). Using the definition of $P^k_1$ and $P^{k,\bar{v}}_2$, we then have that for all $\bar{j} \in S_o$,

$$P^k_1(X = \bar{i}, Y = \bar{u}, \bar{z} = \bar{j}) = P^t_1(k = \bar{i}, u_k = \bar{u}) \cdot p^{k,t}_k(\bar{j} | \bar{i}, \bar{u}),$$

$$P^{k,\bar{v}}_2(\bar{x} = \bar{i}, Y = \bar{u}, Z = \bar{j}) = P^{k,\bar{v}}_{k-1}(k = \bar{i}, u_{k-1} = \bar{u}) \cdot p^{k,t-\tau_0}_{k-1}(\bar{j} | \bar{i}, \bar{u}), \quad \bar{v} \in V(s).$$

This implies that for every $(\bar{i}, \bar{u}) \in R^1_{\bar{u}}$ and every $\bar{v} \in V(s)$,

$$P^k_3(Z = \cdot | X = \bar{i}, Y = \bar{u}) = p^{k,t}_k(\cdot | \bar{i}, \bar{u}), \quad P^{k,\bar{v}}_2(\bar{x} = \bar{i}, Y = \bar{u}) = p^{k,t-\tau_0}_{k-1}(\cdot | \bar{i}, \bar{u}).$$ 

Then, since $P^k_3 = (1 - \gamma)P^k_1 + \gamma \sum_{\bar{v} \in V(s)} \bar{p}_s(\bar{v})P^{k,\bar{v}}_2$, using Eqs. (A.14), (A.15) and the relation $P^k_3(Z = \cdot | X = \bar{i}, Y = \bar{u}) = P^{k,\bar{v}}_2(\bar{x} = \bar{i}, Y = \bar{u}, Z = \cdot) / P^k_3(X = \bar{i}, Y = \bar{u})$, we obtain

$$p^{k,t+1}_k(\cdot | \bar{i}, \bar{u}) = \left(1 - \gamma\right)P^{k}_1(X = \bar{i}, Y = \bar{u}) \cdot p^{k,t}_k(\cdot | \bar{i}, \bar{u}) + \sum_{\bar{v} \in V(s)} \frac{\gamma \bar{p}_s(\bar{v})P^{k,\bar{v}}_2(X = \bar{i}, Y = \bar{u})}{P^k_3(X = \bar{i}, Y = \bar{u})} \cdot p^{k,t-\tau_0}_{k-1}(\cdot | \bar{i}, \bar{u}).$$ 

(A.16)
This shows that $p_k^{\ell,t+1}(\cdot | \bar{i}, \bar{u})$ is a convex combination of $p_k^{\ell,t}(\cdot | \bar{i}, \bar{u})$ and $p_{k-1}^{\ell,\tau_0}(\cdot | \bar{i}, \bar{u})$, $\bar{v} \in V(s)$. By property (b) of the induction hypothesis,

$$p_k^{\ell,t}(\cdot | \bar{i}, \bar{u}) \in \mathcal{N}_{\delta}(p_{\bar{v}}^{\bar{u}}), \quad p_{k-1}^{\ell,\tau_0}(\cdot | \bar{i}, \bar{u}) \in \mathcal{N}_{\delta}(p_{\bar{v}}^{\bar{u}}), \quad \forall \bar{v} \in V(s).$$

Since the set $\mathcal{N}_{\delta}(p_{\bar{v}}^{\bar{u}})$ is convex, it follows that $p_k^{\ell,t+1}(\cdot | \bar{i}, \bar{u}) \in \mathcal{N}_{\delta}(p_{\bar{v}}^{\bar{u}})$, so it satisfies the requirement in property (b).

Reasoning similarly, and using property (d) of the induction hypothesis, it follows that for all $(\bar{i}, \bar{u}) \in R_{\bar{v}}^1$,

$$p_k^{\ell,t+1}(\cdot | \bar{i}, \bar{u}) = p_{\bar{v}}^{\bar{u}}, \quad \forall k \geq k_{t+1}. $$

So $p_k^{\ell,t+1}$ satisfies the requirement in property (d) for $t + 1$.

We now verify that $\mu_k^{\ell,t+1}$ satisfies the requirements in properties (b) and (d) for $t + 1$. Similar to the preceding proof, for each $\bar{i} \in S_o$, either $\mu_k^{\ell,t+1}(\cdot | \bar{i}) = \mu(\cdot | \bar{i})$ (when $P_k^{\ell}(X = \bar{i}) = 0$), or it can be expressed as a convex combination of $\mu_k^{\ell,t}(\cdot | \bar{i})$ and $\mu_{k-1}^{\ell,\tau_0}(\cdot | \bar{i})$, $\bar{v} \in V(s)$:

$$\mu_k^{\ell,t+1}(\cdot | \bar{i}) = \frac{(1 - \gamma) P_k^{\ell}(X = \bar{i})}{P_k^{\ell}(X = \bar{i})} \cdot \mu_k^{\ell,t}(\cdot | \bar{i}) + \sum_{\bar{v} \in V(s)} \gamma \mathbb{P}_\bar{u}(\bar{v}) \frac{P_k^{\ell}(X = \bar{i})}{P_k^{\ell}(X = \bar{i})} \cdot \mu_{k-1}^{\ell,\tau_0}(\cdot | \bar{i}).$$

It then follows from properties (b) and (d) of the induction hypothesis that $\text{supp}(\mu_k^{\ell,s+1}(\cdot | \bar{i})) \subset U(\bar{i})$ for all $\bar{i} \in S_o$, and $\mu_k^{\ell,s+1} = \mu$ for $k \geq k_{t+1}$, which are the requirements in properties (b) and (d).

**For $k = 1$:** The arguments in this case are similar to those for $k \geq 2$. We start with the same definitions. Let $P_1^1$ denote the law of $(i_1, u_1, i_2)$ under $P_{\bar{v}}^{\bar{u}}$, and for each $\bar{v} \in V(s)$, let $P_{\bar{v}}^{\ell,\tau_0}$ denote the law of $(i_0, u_0, i_1)$ under $P_{\bar{v}}^{\bar{u}}$. Let $P_3^1$ denote the convex combination of them, given by Eq. (A.13). Define the random variables $X, Y$ and $Z$ on the sample space $\Omega = S_o \times U \times S_o$ as in the preceding case of $k \geq 2$. Let $I[\cdot \in \cdot]$ denote the indicator function which takes the value 1 if the expression inside $[\cdot \in \cdot]$ is true and takes the value 0 otherwise. Since for every $\bar{v} \in V(s)$,

$$P_{2}^{\ell,\tau}(X = \bar{i}, Y = \bar{u}) = P_{\bar{v}}^{\bar{u}}(i_0 = \bar{i}, u_0 = \bar{u}) = I[\bar{i} \in s, \bar{u} = \bar{u}], \quad (\bar{i}, \bar{u}) \in R_{\bar{v}}^1,$$

we have

$$P_{3}^{\ell,\tau}(X = \bar{i}, Y = \bar{u}) = (1 - \gamma) P_{1}^{\ell}(X = \bar{i}, Y = \bar{u}) + \gamma I[\bar{i} \in s, \bar{u} = \bar{u}], \quad (\bar{i}, \bar{u}) \in R_{\bar{v}}^1. \quad (A.17)$$

Notice that $\text{supp}(P_{3}^{\ell,\tau}) \subset R_{\bar{v}}^1 \times S_o$ because $(s, \bar{u}) = (j_1^1, u_1^1) \in R_{\bar{v}}^1$, and $\text{supp}(P_{1}^{\ell,\tau}) \subset R_{\bar{v}}^1 \times S_o$ by property (b) of the induction hypothesis (in particular, the property of $\mu_1^{\ell,\tau}$). Hence we can write the term $C_1$ in Eq. (A.4) as

$$C_1 = \sum_{i \in S_o} \sum_{\bar{u} \in U(\bar{i})} P_{3}^{\ell,\tau}(X = \bar{i}, Y = \bar{u}) \cdot g_1^{\ell,t+1}(\bar{i}, \bar{u}), \quad (A.18)$$

where $g_1^{\ell,t+1}(\bar{i}, \bar{u})$ for every $(\bar{i}, \bar{u})$ is defined as: if $P_{3}^{\ell,\tau}(X = \bar{i}, Y = \bar{u}) = 0$, then $g_1^{\ell,t+1}(\bar{i}, \bar{u}) = g_0(\bar{i}, \bar{u})$; otherwise,

$$g_1^{\ell,t+1}(\bar{i}, \bar{u}) = \frac{(1 - \gamma) P_{3}^{\ell,\tau}(X = \bar{i}, Y = \bar{u})}{P_{3}^{\ell,\tau}(X = \bar{i}, Y = \bar{u})} \cdot g_1^{\ell,t+1}(\bar{i}, \bar{u}) + \gamma I[\bar{i} \in s, \bar{u} = \bar{u}] \cdot \sum_{\bar{v} \in V(s)} \mathbb{P}_{\bar{u}}(\bar{v}) \cdot g_0^{\ell,\tau_0}(s, \bar{u}, \bar{v}), \quad (A.19)$$

which, for $(\bar{i}, \bar{u}) \neq (s, \bar{u})$, is $g_1^{\ell,t+1}(\bar{i}, \bar{u}) = g_1^{\ell,t}(\bar{i}, \bar{u})$. 

36
We verify that $g_{t+1}^{f,t}$ satisfies the requirement in property (c) for $t + 1$: $g_{t+1}^{f,t} \in \mathcal{N}_\delta(g_\nu)$. By the definition of $\mathcal{N}_\delta(g_\nu)$, what we need to show is that for each $(\tilde{i}, \tilde{u}) \in R^1_{\nu}$,

$$\left|g_{t+1}^{f,t}(\tilde{i}, \tilde{u}) - g_{t}^{f,t}(\tilde{i}, \tilde{u})\right| \leq \delta, \quad \text{and} \quad g_{t+1}^{f,t}(\tilde{i}, \tilde{u}) = 0 \quad \text{if} \quad \tilde{i} = 0.$$  \hfill (A.20)

From the definition of $g_{t+1}^{f,t}$ and the fact that $g_{t}^{f,t} \in \mathcal{N}_\delta(g_\nu)$ [property (c) of the induction hypothesis], we see that Eq. (A.20) is obviously true for all $(\tilde{i}, \tilde{u}) \neq (s, \tilde{u})$ and for the case where $g_{t+1}^{f,t}(\tilde{i}, \tilde{u}) = g_{t}^{f,t}(\tilde{i}, \tilde{u})$. This leaves us only one case to consider: $(\tilde{i}, \tilde{u}) = (s, \tilde{u})$ and $g_{t+1}^{f,t}(s, \tilde{u})$ is given by Eq. (A.19).

By Eq. (A.19), $g_{t+1}^{f,t}(s, \tilde{u})$ is a convex combination of $g_{t}^{f,t}(s, \tilde{u})$ and $\sum_{\tilde{v} \in V(s)} \tilde{v} \cdot \gamma \geq 0$. This proves that $g_{t+1}^{f,t}(0, 0) = 0$ as desired. Consider now the case $s \neq 0$. By property (c) of the induction hypothesis,

$$g_{t+1}^{f,t}(s, \tilde{u}) \in \mathcal{N}_\delta(g_\nu), \quad \forall \tilde{v} \in V(s).$$

Since $g_{t}^{f,t}(s, \tilde{u}) = \sum_{\tilde{v} \in V(s)} \tilde{v} \cdot \gamma \geq 0$, this implies that

$$\left|g_{t+1}^{f,t}(s, \tilde{u}) - g_{t}^{f,t}(s, \tilde{u})\right| \leq \delta, \quad \text{if} \quad s \neq 0.$$  \hfill (A.21)

Combining the preceding relations with the induction hypothesis that $g_{t}^{f,t} \in \mathcal{N}_\delta(g_\nu)$, we have

$$\left|g_{t+1}^{f,t}(s, \tilde{u}) - g_{t}^{f,t}(s, \tilde{u})\right| \leq \delta, \quad \text{and} \quad g_{t+1}^{f,t}(s, \tilde{u}) = 0 \quad \text{if} \quad s = 0,$$

which is Eq. (A.20) for $(\tilde{i}, \tilde{u}) = (s, \tilde{u})$. This proves that $g_{t+1}^{f,t} \in \mathcal{N}_\delta(g_\nu)$, which is the requirement in property (c).

We define the transition probability distributions $\mu_{t+1}^{f,t}$, $p_{t+1}^{f,t}$ by Eqs. (A.13), (A.14), respectively, for $k = 1$, so that Eqs. (A.10), (A.11) hold for $k = 1$ as well. Evidently $\mu_{t+1}^{f,t}$ satisfies the requirement in property (b) for $t + 1$, because supp($p_{t}^{f,t}$) $\subset R^1_{\nu} \times S_\nu$ as discussed earlier. We now verify that $p_{t+1}^{f,t}$ satisfies the requirement in property (b) for $t + 1$, namely,

$$p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) \in \mathcal{N}_\delta(p_{\tilde{u}}), \quad \forall (\tilde{i}, \tilde{u}) \in R^1_{\nu}. \hfill (A.22)$$

Similar to the analysis given earlier for the case $k = 0$, we have that for every $(\tilde{i}, \tilde{u}) \in R^1_{\nu}$, either $p_{t+1}^{f,t}(X = \tilde{i}, Y = \tilde{u}) = 0$ and $p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) = p_{\tilde{u}}$ by definition, or $p_{t+1}^{f,t}(X = \tilde{i}, Y = \tilde{u}) > 0$ and $p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u})$ can be expressed as the convex combination

$$p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) = \frac{(1 - \gamma)}{p_{t}^{f,t}(X = \tilde{i}, Y = \tilde{u})} \cdot p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) + \gamma \sum_{\tilde{v} \in V(s)} \tilde{v} \cdot \gamma \geq 0.$$  \hfill (A.23)

For $(\tilde{i}, \tilde{u}) \neq (s, \tilde{u})$, Eq. (A.23) is $p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) = p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u})$; since $p_{t+1}^{f,t}(\cdot | \tilde{i}, \tilde{u}) \in \mathcal{N}_\delta(p_{\tilde{u}})$ by property (b) of the induction hypothesis, to prove Eq. (A.23), we only have one case left to consider: $(\tilde{i}, \tilde{u}) = (s, \tilde{u})$ and $p_{t+1}^{f,t}(\cdot | s, \tilde{u})$ is given by Eq. (A.22). Now if $s = 0$, then $(\tilde{i}, \tilde{u}) = (s, \tilde{u}) = (0, 0)$ and we have $p_{t+1}^{f,t}(0 | 0, 0) = 1$ as desired, because $p_{t+1}^{f,t}(0 | 0, 0) = 1$ by the induction hypothesis and $p_{t+1}^{f,t}(0 | 0, 0) = 1$ by our definition of $p_{t+1}^{f,t}$ for $s = 0$. So consider the case $(\tilde{i}, \tilde{u}) = (s, \tilde{u}) \neq (0, 0)$. By property (b) of the induction hypothesis,

$$p_{t+1}^{f,t}(\cdot | s, \tilde{u}, \tilde{v}) \in \mathcal{N}_\delta(p_{\tilde{u}}), \quad \forall \tilde{v} \in V(s). \hfill (A.23)$$
In view of Eq. (A.22) in the definition of SSP($\bar{\nu}$) [Definition A.1], $p_{0}^{\tilde{\mu}} = \sum_{\tilde{\nu} \in V(s)} p_{s}(\tilde{\nu}) p_{0}^{\tilde{\mu}}$, and therefore, the relation (A.23) implies that

$$\sum_{\tilde{\nu} \in V(s)} \tilde{\nu}_{s}(\tilde{\nu}) p_{0}^{\tilde{\mu}}(\cdot | s, \tilde{\nu}, \tilde{\nu}) \in \tilde{N}_{\delta}(p_{0}^{\tilde{\mu}}).$$

Using this fact and the induction hypothesis that $p_{t+1}^{\tilde{\mu}}(\cdot | s, \tilde{\nu}) \in \tilde{N}_{\delta}(p_{0}^{\tilde{\mu}})$, we obtain from the convex combination formula (A.22) that $p_{t+1}^{\tilde{\mu}}(\cdot | s, \tilde{\nu}) \in \tilde{N}_{\delta}(p_{0}^{\tilde{\mu}})$. This proves Eq. (A.21) and shows that $p_{t+1}^{\tilde{\mu}}$ satisfies the requirement in property (b) for $t+1$.

Define the Markov chain for $t+1$:

We now define the time-inhomogeneous Markov chain $(i_0, u_0, v_0, (i_1, u_1), (i_2, u_2), \ldots)$, with probability distribution $P_{t+1}^{\ell}$, as required in property (a) for $t+1$. Let the chain start with $(i_0, u_0, v_0) = (i, u, v)$, and let its transition probabilities have the product forms given in property (b) for $t+1$, where $P_{k}^{\ell+1}, k \geq 0$, and $P_{k}^{\ell+1}, k \geq 1$, are the functions that we defined in the preceding proof. Also let the time-varying one-stage cost functions $g_{k}^{\ell+1}, k \geq 0$, be as defined earlier. We have shown that these transition probabilities and one-stage cost functions satisfy the requirements in properties (b)-(d). To prove the lemma, what we still need to show is that with our definitions, the expression given in property (a) equals $\tilde{Q}_{t+1}(i, u, v)$.

First of all, our definitions of the transition probabilities and one-stage cost functions for time $t+1$ ensure that $(i_k, u_k, v_k, \ldots)$ evolves and incurs costs as in SSP($\bar{\nu}$) under the proper distribution $\mu$ [property (d)]. Consequently, $E^{P_{t+1}^{\ell}}\left[\sum_{k=1}^{\infty} g_{k}^{\ell+1}(i_k, u_k)\right]$ is well-defined and finite, and the order of summation and expectation can be exchanged:

$$E^{P_{t+1}^{\ell}}\left[\sum_{k=1}^{\infty} g_{k}^{\ell+1}(i_k, u_k)\right] = \sum_{k=1}^{\infty} E^{P_{t+1}^{\ell}}\left[g_{k}^{\ell+1}(i_k, u_k)\right].$$

Now $\tilde{Q}_{t+1}(i, u, v) = \sum_{k=0}^{\infty} C_{k}$ by Eq. (A.1). Hence, to prove property (a) for $t+1$, that is, to show

$$\tilde{Q}_{t+1}(i, u, v) = g_{0}^{\ell+1}(i, u, v) + \sum_{k=1}^{\infty} E^{P_{t+1}^{\ell}}\left[g_{k}^{\ell+1}(i_k, u_k)\right],$$

we only need to show that

$$C_{0} = g_{0}^{\ell+1}(i, u, v), \quad C_{k} = E^{P_{t+1}^{\ell}}\left[g_{k}^{\ell+1}(i_k, u_k)\right], \quad k \geq 1. \quad (A.24)$$

The equality for $C_{0}$ above is true since by definition $g_{0}^{\ell+1}(i, u, v) = g_{t+1}(i, u, v) = C_{0}$ [cf. Eq. (A.6)]. We now prove the second equality in Eq. (A.24) for $C_{k}, k \geq 1$.

For $k \geq 1$, recall

$$C_{k} = \sum_{\bar{i}, \bar{u} \in U(\bar{i})} P_{3}^{k}(X = \bar{i}, Y = \bar{u}) \cdot g_{k}^{\ell+1}(\bar{i}, \bar{u})$$

[cf. Eqs. (A.9), (A.18)]. Hence, to show the desired equality (A.24) for $C_{k}$, it is sufficient to show that

$$P_{t+1}^{\ell}(i_k = \bar{i}, u_k = \bar{u}) = P_{3}^{k}(X = \bar{i}, Y = \bar{u}), \quad \forall (\bar{i}, \bar{u}) \in R_{3}^{k}. \quad (A.25)$$

By the definition of $P_{t+1}^{\ell}$ [which is defined by property (b), as we recall], $P_{t+1}^{\ell}(u_k = \bar{u} | i_k = \bar{i}) = \mu_{k}^{\ell+1}(\bar{u} | \bar{i})$ for all $(\bar{i}, \bar{u}) \in R_{3}^{k}$, so in view of Eq. (A.10) (which is the defining relation for $\mu_{k}^{\ell+1}$), the equality (A.25) will be implied if we show

$$P_{t+1}^{\ell}(i_k = \bar{i}) = P_{3}^{k}(X = \bar{i}), \quad \forall \bar{i} \in S_{0}. \quad (A.26)$$
We verify Eq. (A.26) by induction on $k$. For $k = 1$, from Eq. (A.17) and property (b) of $P^\ell_t$, we have that for every $\bar{i} \in S_o$,

\[
P^1_3(X = \bar{i}) = (1 - \gamma) P^\ell_t(i_1 = \bar{i}) + \gamma e_s(\bar{i})
\]

\[
= (1 - \gamma) P^\ell_t(i_1 = \bar{i}) + \gamma e_s(\bar{i})
\]

\[
= (1 - \gamma) q^\ell_t(\bar{i}) + \gamma e_s(\bar{i})
\]

\[
= q^\ell_{t+1}(\bar{i}) = P^\ell_t(i_1 = \bar{i}),
\]

where the last three equalities follow from the definition of $q^\ell_{t+1}$ [cf. Eq. (4.9)], the definition of $p^\ell_0_{t+1}$ [Eq. (A.7)], and the definition of $P^\ell_{t+1}$, respectively. Hence Eq. (A.26) holds for $k = 1$.

Suppose Eq. (A.26) holds for some $k \geq 1$. Then, by the definition of $P^\ell_{t+1}$ [i.e., the property (b)], we have for all $\bar{j} \in S_o$,

\[
P^\ell_{t+1}(i_{k+1} = \bar{j}) = \sum_{\bar{i} \in S_o} \sum_{\bar{u} \in U(\bar{i})} P^\ell_{t+1}(i_k = \bar{i}) \cdot \mu^\ell_{k+1}(\bar{u} | \bar{i}) \cdot \mu^\ell_{k+1}(\bar{j} | \bar{i}, \bar{u})
\]

\[
= \sum_{\bar{i} \in S_o} \sum_{\bar{u} \in U(\bar{i})} P^k_3(X = \bar{i}) \cdot \mu^\ell_{k+1}(\bar{u} | \bar{i}) \cdot \mu^\ell_{k+1}(\bar{j} | \bar{i}, \bar{u})
\]

\[
= P^k_3(Z = \bar{j}) = P^{k+1}_3(X = \bar{j}),
\]

where the second equality follows from the induction hypothesis, the third equality follows from Eqs. (A.10)-(A.11), and the last equality follows from the definition of $P^k_3$ and $P^{k+1}_3$. This completes the induction and proves that Eq. (A.26) holds for all $k \geq 1$, which in turn proves that Eq. (A.26) holds for all $k \geq 1$. Consequently, for all $k \geq 1$, the desired equality (A.24) for $C_k$ holds. This completes the proof of Lemma 4.5.