The Shannon Switching Game: Epsilon-Greedy Q-Learning

Albert Ding, Cal Hartzell, Chas Huang, Calvin Kuo

Introduction to Algorithmic Economics – Spring 2023

University of Virginia

Proof of Termination Time

We show that the game terminates in $O(m)$ iterations where m is the number of edges in the graph.

Lemma 1.1

No games will have an unsecured edge in the graph after iteration $\lceil \frac{m}{2} \rceil$.

Proof. Consider any finite graph $G = (V, E)$. m, the number of edges in the graph, is equal to $|E|$, the cardinality of the set of edges E.

On each iteration, the fix-type player secures an unsecured edge and the cut-type player deletes an unsecured edge. Since on each iteration, two unsecured edges are secured or deleted, the maximum number of iterations it can take for all of the edges to either be secured or deleted is $\lceil \frac{m}{2} \rceil$ iterations, the value obtained when $\frac{m}{2}$ is rounded up to the nearest integer. Thus, there can be no unsecured edges remaining after iteration $\lceil \frac{m}{2} \rceil$.

Lemma 1.2

All games will terminate on or before iteration $\lceil \frac{m}{2} \rceil$.

Proof. We can partition the set of games into two subsets: those that terminate before iteration $\lceil \frac{m}{2} \rceil$, and those that terminate on or after iteration $\lceil \frac{m}{2} \rceil$. We will show that all of the games in the latter subset will terminate on iteration $\lceil \frac{m}{2} \rceil$ by showing that one of the termination conditions must be satisfied.

Consider the two termination conditions, "a secured path exists from s to t" and "no paths exist between s and t". By Lemma 1.1, all of the edges must be secured or deleted by iteration $\lceil \frac{m}{2} \rceil$, with no unsecured edges remaining. Thus, at this point in the game, the condition of "a secured path exists from s to t" is equivalent to "a path (at least one path) exists from s to t". Since the negation of "at least one path exists from s to t" is "no paths exist between s and t", by the law of the excluded middle, either one or the other termination condition must be true.

Thus, any game that has not terminated before iteration $\lceil \frac{m}{2} \rceil$ must terminate on iteration $\lceil \frac{m}{2} \rceil$. Since all games either terminate before iteration $\lceil \frac{m}{2} \rceil$ or terminate on iteration $\lceil \frac{m}{2} \rceil$, all games terminate within $\lceil \frac{m}{2} \rceil$ iterations.

By Lemma 1.2, all games terminate within $\lceil \frac{m}{2} \rceil$ iterations. Applying the limit definition of $O(m)$, $\lim_{m\to\infty} \frac{\lceil \frac{m}{2} \rceil}{m} = \frac{1}{2}$. Since $\frac{1}{2} \in [0, \infty)$, $\lceil \frac{m}{2} \rceil \in O(m)$. Thus, the game will terminate in $O(m)$ iterations. □

The Switching Game

Proposed by Claude Shannon, the switching game is a two-player game over a finite graph $G = (V, E)$ where V is the set of vertices, and E is the set of edges. There are two target vertices in the graph, denoted by $s, t \in V$. The two players are of two static types: fix and cut. Initially, all edges are in the state of "unsecured". The game proceeds in discrete time, where players make moves alternatively:

(i) the goal of the fix-type player is to secure a path from s to t
(ii) the goal of the cut-type player is to disconnect s and t.

Specifically, in each iteration,

(i) the fix-type player secures an edge in the graph, and
(ii) the cut-type player deletes an unsecured edge in the graph.

The game ends when there is a secured path from s to t (i.e., fix wins) or there are no paths between s and t (i.e., cut wins).

Epsilon-Greedy Q-Learning

Epsilon-greedy Q-learning is a reinforcement learning algorithm that is used for solving problems where an agent has to make decisions in an environment in order to maximize a long-term reward.

In epsilon-greedy Q-learning, the agent maintains a table of Q-values, where each entry represents the expected reward for taking a particular action in a particular state. At each time step, the agent selects an action based on the current Q-values. The agent either takes the action that has the highest Q-value (i.e., the greedy action) or a random action with probability ϵ.

- The value of ϵ is typically set to a small value, such as 0.1 or 0.2, which means that the agent will exploit its current knowledge most of the time but occasionally explore new actions. This helps the agent to discover better strategies in the long run while still exploiting the best strategy that it knows.
- The Q-values are updated using the Bellman equation, which states that the expected reward for taking an action in a state is equal to the immediate reward plus the discounted expected reward for the next state-action pair. The discount factor is used to account for the fact that future rewards are less valuable than immediate rewards.

References

Baeldung. Epsilon-greedy q-learning. 2023.
URL: `https://www.baeldung.com/cs/epsilon-greedy-q-learning`.

Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948.

Example Games

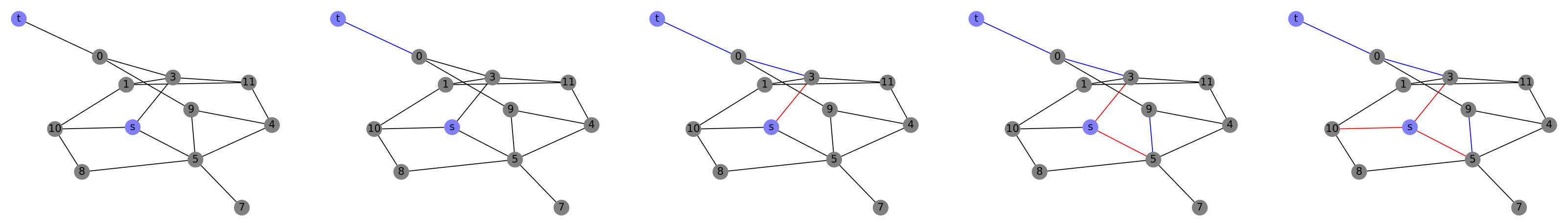

Figure 1. A cut-type player wins against the trained fix-type agent.

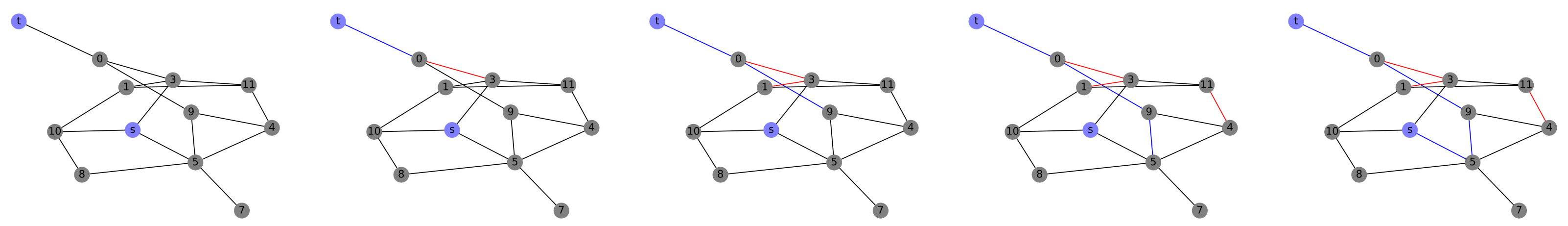

Figure 2. A fix-type player wins against the trained cut-type agent.

Epsilon-Greedy Q-Learning Performance

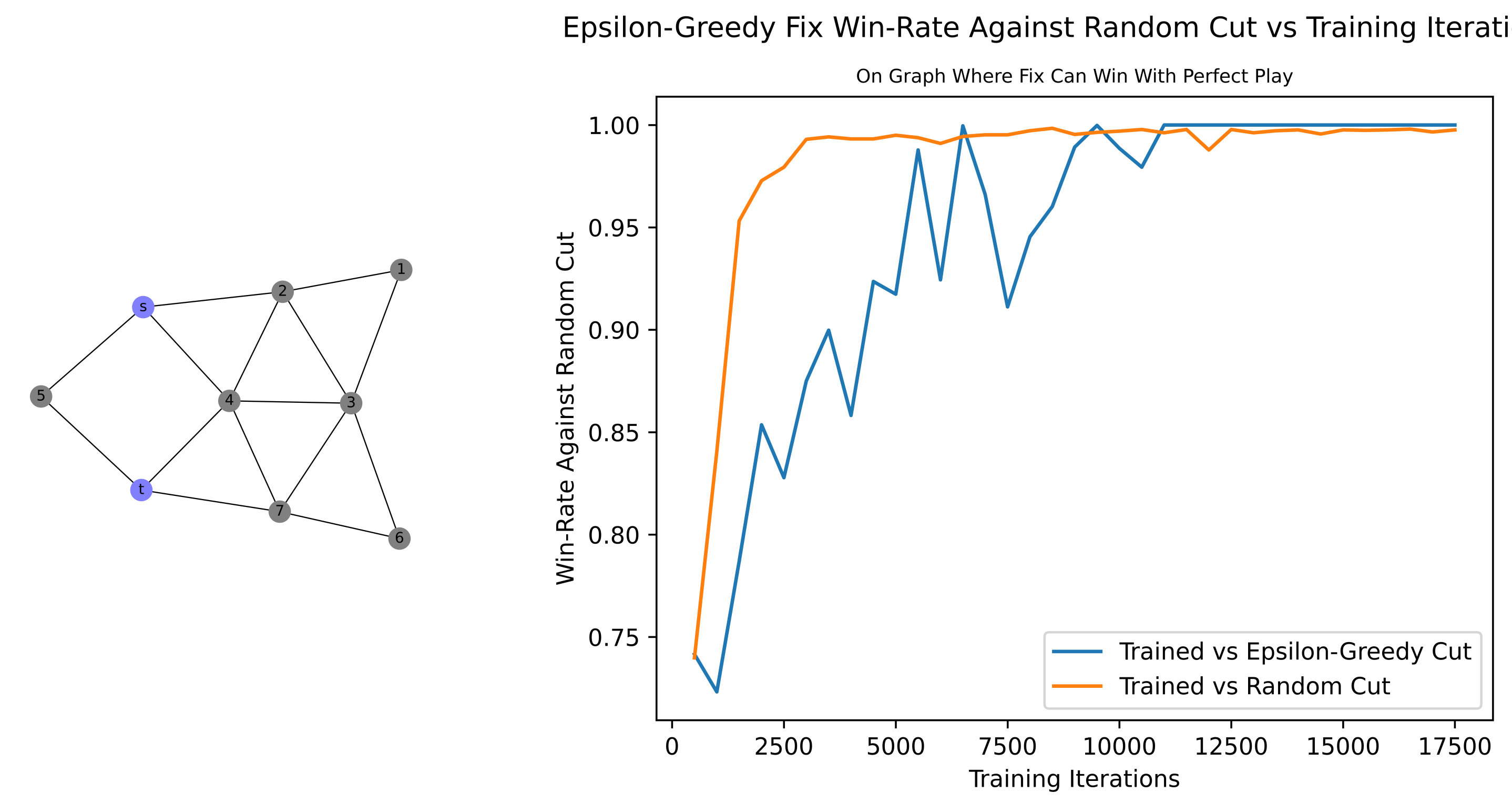

Figure 3. Plot showing improvement in epsilon-greedy fix win-rate as iterations increase and the associated graph where fix can always win with perfect play. $\epsilon = 0.2$, learning rate = 0.5, discount factor = 0.8, 500 games against random cut after each 500 iterations, averaged over 10 trials.

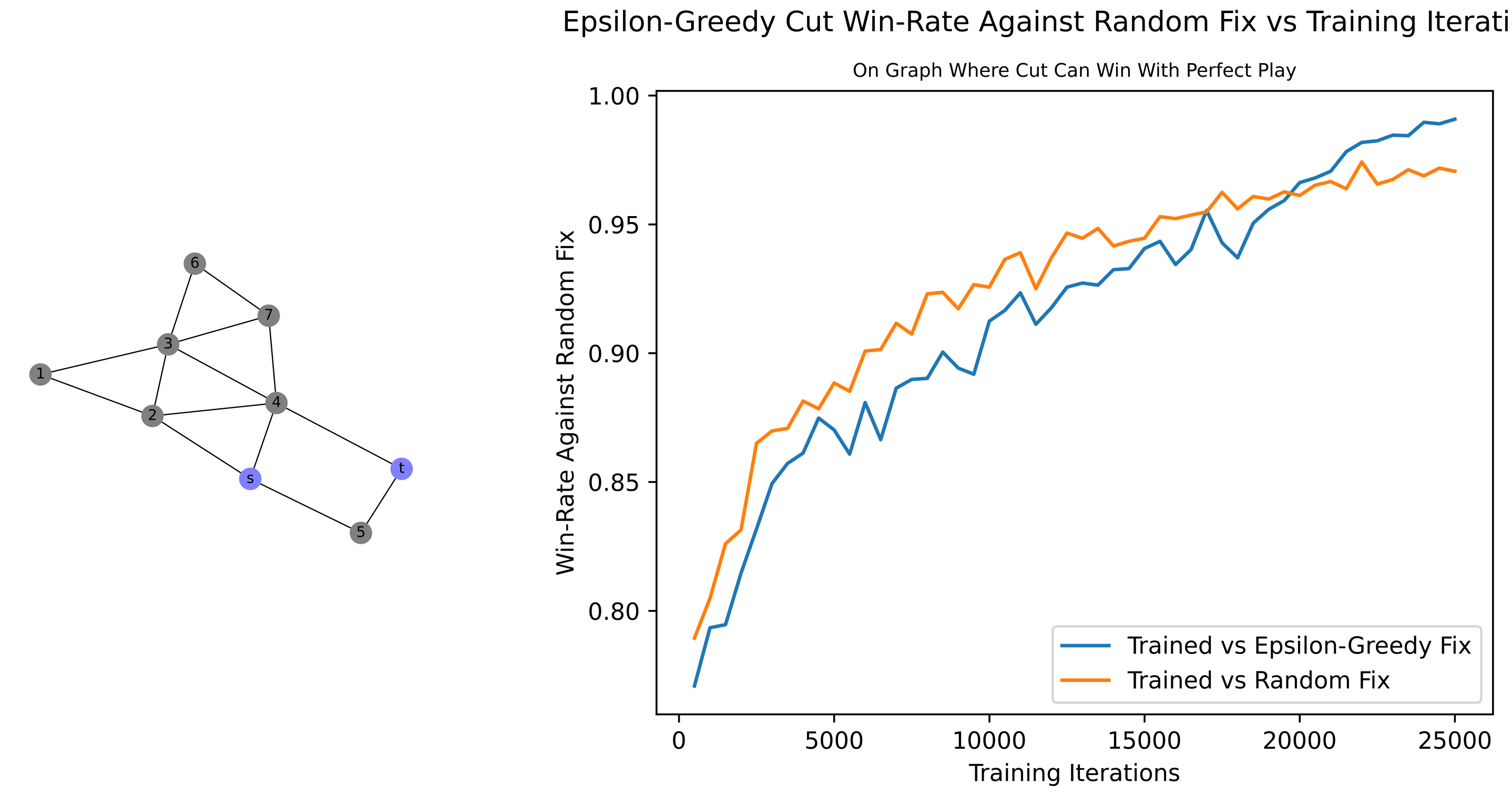

Figure 4. Plot showing improvement in epsilon-greedy cut win-rate as iterations increase and the associated graph where cut can always win with perfect play. $\epsilon = 0.2$, learning rate = 0.5, discount factor = 0.8, 500 games against random fix after each 500 iterations, averaged over 10 trials.