Regularized Minimax-V Learning for Solving Randomly Terminating Two-player Zero-sum Markov Games

Xinxiang Guo[1,2][0000−0003−4315−1063] and Yifen Mu*[2,3][0000−0001−6478−6928]

[1] School of Mathematical Sciences, University of Chinese Academy of Sciences
[2] Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences
[3] State Key Laboratory of Mathematical Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences

Abstract. In this paper, we investigate randomly terminating two-player zero-sum Markov games. This game model differs from infinite-horizon discounted zero-sum Markov games and can be used to model many practical scenarios; however, related work on such games is still insufficient. We propose the regularized minimax-V learning algorithm and prove that the value sequence generated by this algorithm converges to the minimax value function with an appropriate choice of regularization parameter sequence. This algorithm applies the Euclidean regularization technique to accelerate the convergence in a different way compared with previous literature. Through simulation experiments, we demonstrate the convergence of regularized minimax-V learning algorithm in the ratio game and a randomly generated Markov game. To the best of our knowledge, for randomly terminating two-player zero-sum Markov games, this paper presents the first accelerated NE-solving algorithm.

Keywords: Markov game · Euclidean regularization · V-value iteration · Accelerating technique.

1 Introduction

In recent years, multi-agent reinforcement learning [3] has found widespread applications in modeling interactions among multiple agents or between agents and the environment. Specific applications include real-time strategy games [29], chess and card games [21, 26], robot control [17], energy networks [10] and so on. In game theory, multi-agent reinforcement learning can be framed as stochastic games [20, 25]. Depending on the relationship among agents, stochastic games are categorized into cooperative [30], competitive [20], and mixed settings [34]. Competitive settings in stochastic games often refer to two-player zero-sum Markov games (MGs). Specifically, one variety is the randomly terminating MG, which can be used to model numerous scenarios, such as robot sports, gaming, racing, and so on. However, most literature focuses on infinite-horizon MGs with

* Corresponding author: mu@amss.ac.cn

a discount parameter, and research on randomly terminating MGs is relatively scarce [25]. This paper considers the randomly terminating MGs and investigates methods of finding Nash equilibria for such games.

Previous equilibrium-solving algorithms can be categorized into two types: value-based methods and policy-based methods. Policy-based methods [1, 35] typically parameterize policies and update these parameters directly using gradient information. Such methods include REINFORCE [32], actor-critic [18], extra-gradient [19] algorithms, and so on. Despite their advantage of scalability to large action spaces through function approximation, policy-based methods often face challenges in directly obtaining gradients.

Value-based methods maintain a value sequence to estimate the value of a state or state-action pair and iteratively update these estimates. Most value-based methods iterate over the value of state-action pairs (i.e., Q-values) and generate the current policy using the updated Q-values [20, 24]. Beyond value-based methods that iteratively update Q-value functions, there are also value-based methods that iteratively update the state value function (i.e., V-values) [16, 25]. Compared to iteratively updating the Q-value function, iteratively updating the V-value function is simpler and can still be theoretically guaranteed to learn a Nash equilibrium under certain conditions. The algorithms proposed in this paper fall into this class.

Recently, research has extended beyond focusing solely on the convergence of algorithms to Nash equilibria to also include considerations of convergence efficiency. This encompasses the convergence conditions [4, 14], the convergence rate [6, 9], and the convergence mode (i.e., time-average or last-iterate) [13]. Regularization techniques are simple yet effective methods that balance exploitation and exploration by introducing a regularization term into the objective function. This technique finds wide applications in fields such as machine learning, statistical learning, online learning, game theory, and multi-agent reinforcement learning.

The main motivation of this paper is to study the randomly terminating two-player zero-sum MGs and propose an accelerated algorithm using regularization techniques without losing convergence. We propose a regularized minimax-V learning algorithm to solve the NE and introduce Euclidean regularization to accelerate the process of solving a saddle point problem. We prove the convergence of the value sequence generated by this algorithm to the minimax value function with an appropriate choice of the regularization parameter sequence. Through simulation experiments, we demonstrate the convergence of the regularized minimax-V learning algorithm in the ratio game and a randomly generated Markov game.

1.1 Related work

Two-player zero-sum Markov games (MGs). According to the time horizon, two-player zero-sum MGs can be classified into three categories: infinite-horizon [4, 31, 33], episodic finite-horizon [2, 16], and randomly terminating horizon. Compared with the first two categories, research on randomly terminating

MGs is relatively lacking. Shapley [25] first introduced this game model and proposed an algorithm analogous to value iteration. Daskalakis et al. [8] showed that if both players run policy gradient methods in tandem, their policies will converge to a min-max equilibrium of the game, as long as their learning rates follow a necessary two-timescale rule. Policy-based learning algorithms usually require additional conditions on the game model to achieve convergence. This paper also considers the randomly terminating version and proposes two algorithms that converge to the NE without additional conditions.

The regularized minimax-V learning algorithm is inspired by the work of Littman [20]. Compared with the V-learning algorithm [16], which also updates V-values at each iteration, this paper does so in a completely different manner. The V-learning algorithm is designed for finite-horizon episodic MGs, where the value sequence can be updated from the last step to the first step within each episode. However, since the horizon of randomly terminating MGs is potentially infinite, V-learning is no longer applicable.

Regularization technique for learning in games. Generally speaking, there are two main approaches to applying regularization techniques in game learning. One approach is to incorporate the regularization term into the objective function of the learning algorithm, which can improve its performance [5, 27]. The other approach is to integrate the regularization term into the payoff information [23]. Cen et al. [7] use entropy regularization to modify the original matrix game, resulting in an algorithm that can find an approximate Nash equilibrium at a sublinear rate. In our work, we utilize the Euclidean norm as the regularization term to attain a faster convergence rate, and the entropy regularization does not apply here because the entropy of a probability distribution is unbounded.

Paper structure: in Section 2, we introduce the randomly terminating two-player zero-sum Markov game model and the definition of norm used later; in Section 3, we propose the regularized minimax-V learning algorithm, prove its convergence to the minimax-value function; in Section 4, we implement several simulation experiments including ratio game and randomly generated Markov game; in Section 5, we conclude this paper and introduce some possible future directions.

2 Problem Formulation

2.1 Game model

The randomly terminating two-player zero-sum Markov game (MG) was proposed by Shapley [25] and can be represented by a tuple $(\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{B}, r, p, \zeta)$ with each element defined as follows:

- $\mathcal{N} := \{X, Y\}$ is the set of players, i.e., the participants of the game are player X and Y.
- $\mathcal{S}$ is the finite state space.

- $\mathcal{A}$ is the finite action set for player X and $\mathcal{B}$ is the finite action set for player Y.
- $r : \mathcal{S} \times \mathcal{A} \times \mathcal{B} \to [-1, 1]$ is the payoff function for player Y and the cost function for player X.
- $p : \mathcal{S} \times \mathcal{A} \times \mathcal{B} \to \mathcal{S}$ is the state transition function, i.e., $p(s'|s, a, b)$ gives the probability of transmitting from state s to state s' if player X takes action a and player Y takes action b.
- $\zeta := \min_{(s,a,b)\in\mathcal{S}\times\mathcal{A}\times\mathcal{B}}(1 - \sum_{s'} p(s'|s, a, b)) > 0$ is the lower bound on the probability that the MG will terminate at some state after both players take some actions.

A stationary policy (aka. Markovian policy) for player X refers to a set of mappings denoted by $\Delta(\mathcal{A})^{|\mathcal{S}|} \ni \mathbf{x} := \{x_s\}_{s\in\mathcal{S}}$ where $x_s : \mathcal{S} \to \Delta(\mathcal{A})$ maps each state to an action distribution. The stationary policy $\mathbf{y}$ and mapping y_s for player Y can be similarly defined.

For a pair of stationary policies $(\mathbf{x}, \mathbf{y})$ and an initial state s, the value that player Y gains and player X pays can be represented as

$$\begin{aligned} V^{\mathbf{x},\mathbf{y}}(s) = \mathbb{E}[\sum_{t=0}^{T} r(s_t, a_t, b_t)|s_0 = s, a_t \sim x_{s_t}(\cdot), \\ b_t \sim y_{s_t}(\cdot), s_{t+1} \sim p(\cdot|s_t, a_t, b_t), t \geq 0], \end{aligned} \tag{1}$$

where the expectation above is taken with respect to the trajectory induced by the joint policy $(\mathbf{x}, \mathbf{y})$. We call $V^{\mathbf{x},\mathbf{y}} = (V^{\mathbf{x},\mathbf{y}}(s))_{s\in\mathcal{S}}$ *the V-value function*. The minimax game value on state s is then defined as

$$V^*(s) = \min_{x\in\Delta(\mathcal{A})^{|\mathcal{S}|}} \max_{y\in\Delta(\mathcal{B})^{|\mathcal{S}|}} V^{\mathbf{x},\mathbf{y}}(s). \tag{2}$$

We call $V^* = (V^*(s))_{s\in\mathcal{S}}$ *the minimax value function*.

A pair of stationary policies $(\mathbf{x}^*, \mathbf{y}^*)$ is called a *Nash equilibrium* (NE) of the MGs if for any stationary policy $\mathbf{x}$ and $\mathbf{y}$,

$$\begin{aligned} V^{\mathbf{x}^*,\mathbf{y}^*}(s) \leq V^{\mathbf{x},\mathbf{y}^*}(s), \quad \forall s \in \mathcal{S}, \\ V^{\mathbf{x}^*,\mathbf{y}^*}(s) \geq V^{\mathbf{x}^*,\mathbf{y}}(s), \quad \forall s \in \mathcal{S}. \end{aligned} \tag{3}$$

From the definition (2) of minimax game value, it can be easily proven that if a pair of stationary policies $(\mathbf{x}, \mathbf{y})$ satisfies $V^{\mathbf{x},\mathbf{y}}(s) = V^*(s)$ for all s, then the stationary policies $(\mathbf{x}, \mathbf{y})$ is a Nash equilibrium. Moreover, it is known that a pair of stationary policies attaining the minimax value on state s is necessarily attaining the minimax value on all states[11].

Given an initial state s and a pair of stationary policies $(\mathbf{x}, \mathbf{y})$, the *Q-value function* (with respect to player Y) is defined as

$$\begin{aligned} Q_s^{\mathbf{x},\mathbf{y}}(a, b) = \mathbb{E}[\sum_{t=0}^{T} r(s_t, a_t, b_t)|s_0 = s, a_0 = a, b_0 = b, \\ s_{t+1} \sim p(\cdot|s_t, a_t, b_t), a_{t+1} \sim x_{s_{t+1}}(\cdot), \\ b_{t+1} \sim y_{s_{t+1}}(\cdot), t \geq 0] \end{aligned} \tag{4}$$

for all $a \in \mathcal{A}$ and $b \in \mathcal{B}$, where the expectation above is taken with respect to the trajectory induced by the joint policy $(\mathbf{x}, \mathbf{y})$. From their definitions, we know that the V-value function and the Q-value function satisfy

$$Q_s^{\mathbf{x},\mathbf{y}}(a,b) = r(s,a,b) + \sum_{s' \in \mathcal{S}} p(s'|s,a,b) V^{\mathbf{x},\mathbf{y}}(s') \tag{5}$$

and

$$V^{\mathbf{x},\mathbf{y}}(s) = \sum_{a \in \mathcal{A}} \sum_{b \in \mathcal{B}} x_s(a) y_s(b) Q_s^{\mathbf{x},\mathbf{y}}(a,b)$$

for all $s \in \mathcal{S}$, $a \in \mathcal{A}$ and $b \in \mathcal{B}$.

Remark 1. The game defined above is analogous to the infinite-horizon discounted MGs introduced by Littman [20]. In the latter model, the game continues infinitely with a discount parameter γ that diminishes the importance of future rewards. From the perspective of accounting for future reward uncertainty, these two game models share similarities. The randomly terminating version handles uncertainty by allowing the game to potentially terminate with certain probabilities, whereas the infinite-horizon version uses a discount parameter γ less than 1 to address such uncertainties.

For the infinite-horizon discounted MGs, the equation (5) turns to be

$$Q_s^{\mathbf{x},\mathbf{y}}(a,b) = r(s,a,b) + \gamma \sum_{s' \in \mathcal{S}} p(s'|s,a,b) V^{\mathbf{x},\mathbf{y}}(s'), \tag{6}$$

where $\sum_{s' \in \mathcal{S}} p(s'|s,a,b) = 1$. From the aspect of the relation between the V-value and the Q-value function, the infinite-horizon model can be seen as a special case of the randomly terminating model because when we let the state transition probability p in (5) satisfy $\sum_{s' \in \mathcal{S}} p(s'|s,a,b) = \gamma < 1$ for all s, a, b, the relation (6) is obtained.

2.2 Notations

For a matrix $Q^{|\mathcal{A}| \times |\mathcal{B}|}$, its norm is defined to be

$$||Q|| := \max_{a \in \mathcal{A}} \max_{b \in \mathcal{B}} |Q(a,b)| \tag{7}$$

where $Q(a,b)$ is the according element given action profile (a,b). Hence, given two matrices Q_1 and Q_2, their distance is

$$||Q_1 - Q_2|| := \max_{a \in \mathcal{A}} \max_{b \in \mathcal{B}} |Q_1(a,b) - Q_2(a,b)|.$$

Given two V-value functions V_1 and V_2, their distance is defined to be

$$||V_1 - V_2|| := \max_{s \in \mathcal{S}} |V_1(s) - V_2(s)|. \tag{8}$$

3 Regularized Minimax-V Learning

3.1 Contraction operator

In this part, we outline the structure of the regularized minimax-V learning algorithm without introducing the regularization technique.

Let V be a vector in $\mathbb{R}^{|\mathcal{S}|}$ and then $V(s)$ represents the value estimate of state s. Given any vector V, a payoff matrix can be obtained by the formula

$$Q_s(a,b) = r(s,a,b) + \sum_{s' \in \mathcal{S}} p(s'|s,a,b)V(s'), \tag{9}$$

similar to (5). We call the game with payoff matrix Q_s the auxiliary game at state s.

Given the payoff matrix Q_s at state s, denote the value of the according two-player zero-sum game by $f_v(Q_s)$, i.e.,

$$f_v(Q_s) = \min_{x \in \Delta(\mathcal{A})} \max_{y \in \Delta(\mathcal{B})} x^T Q_s y. \tag{10}$$

Then, consider the mapping $M : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ defined by

$$MV(s) = (1-\alpha)V(s) + \alpha f_v(Q_s) \tag{11}$$

where $\alpha \in (0,1)$. For the mapping M, it is easy to prove the following lemma.

Lemma 1. *The mapping $M : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$ is a contraction operator and thus has one unique fixed point. Moreover, the fixed point is the minimax value function of the Markov game.*

Once the fixed point V^* of M is obtained, by solving the NE (x_s^*, y_s^*) under payoff matrix Q_s^* for all state s, we obtain the NE $(\mathbf{x}^*, \mathbf{y}^*)$ of the MGs because the strategy profile $\{(x_s^*, y_s^*)\}_{s \in \mathcal{S}}$ forms the NE strategy of the MGs.

Remark 2. For the fixed point V^*, we have

$$V^* = f_v(Q_s^*), \quad \forall s \in \mathcal{S}. \tag{12}$$

By Lemma 1, the fixed point V^* can be obtained iteratively as below. Starting from any initial point $V_0 \in \mathbb{R}^{|\mathcal{S}|}$ (e.g., $V_0(s) = 0,\ \forall s \in \mathcal{S}$), define

$$V_t = MV_{t-1},\ t \geq 1. \tag{13}$$

Then, the sequence $\{V_t\}_{t=0}^{\infty}$ converges to V^*. When the learning rate α is set to be 0, the formula (11) is just the value iteration format in [25].

3.2 Regularized Minimax-V Learning Algorithm

In this part, we focus on solving the problem (10) and introduce the regularization technique to accelerate the process.

To solve the saddle point problem (10), an approach is to use learning algorithms such as Fictitious Play [12] and Hedge [15]. In the following simulations, we employ the Optimistic Multiplicative Weights Update (OMWU) algorithm [31] as the learning algorithm.

However, the original problem (10) is only convex-concave. Our idea is that by adding an appropriate regularization term, we can transform this problem into a strongly-convex-strongly-concave saddle point problem, and thus accelerate the convergence. Meanwhile, by controlling the regularization parameter, the deviation from the original minimax value is tolerable, and the overall convergence to the minimax value function can be retained.

To be specific, consider the regularized function $f_v^\tau : \mathbb{R}^{|\mathcal{A}|\times|\mathcal{B}|} \to \mathbb{R}$, defined as

$$f_v^\tau(Q) := \min_{x\in\Delta(\mathcal{A})} \max_{y\in\Delta(\mathcal{B})} \left(x^T Q y + \frac{\tau}{2}||x||^2 - \frac{\tau}{2}||y||^2 \right), \tag{14}$$

where $\tau > 0$ is called the regularization coefficient. It is easy to see that the problem (14) is a strongly-convex-strongly-concave saddle point problem.

The difference between $f_v^\tau(Q)$ and $f_v(Q)$ can be proven to be bounded by a constant times of the regularization coefficient. However, this does not apply to entropy regularization since the entropy of a probability distribution is unbounded. Therefore, if we use entropy regularization like [7], the difference between the solutions of the regularized problem and the original problem becomes uncontrollable.

By replacing f_v in (11) with f_v^τ, we obtain the regularized operator $M^\tau : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$, i.e.,

$$M^\tau V(s) = (1-\alpha)V(s) + \alpha f_v^\tau(Q_s), \quad \forall s \in \mathcal{S}, \tag{15}$$

where f_v^τ is defined as (14) and Q_s is defined as (9). For any two value functions V_1 and V_2, by simple calculation, we have

$$||M^\tau V_1 - M^\tau V_2|| \le (1-\alpha\zeta)||V_1 - V_2|| + \alpha\tau C,$$

where $C = \max\{\frac{|\mathcal{A}|-1}{|\mathcal{A}|}, \frac{|\mathcal{B}|-1}{|\mathcal{B}|}\}$. Hence, the operator M^τ is not a contraction operator anymore and we cannot get the minimax value function V^* by directly iteratively letting $V_{t+1} = M^\tau V_t$. Therefore, we make a little modification to (15), the minimax value function can still be reached via the method of value iteration.

Rather than use a constant regularization coefficient, we use a time-varying regularization coefficient τ_t, i.e., for $t = 0, 1, 2, \ldots$,

$$V_{t+1}(s) = (1-\alpha)V_t(s) + \alpha f_v^{\tau_t}(Q_{t,s}), \quad \forall s \in \mathcal{S}, \tag{16}$$

where the initial condition V_0 can be arbitrarily set. Usually, we would set $V_0(s) = 0$ for all $s \in \mathcal{S}$.

Algorithm 1: Regularized Minimax-V Learning Algorithm

Input : Learning rate $\alpha \in (0,1)$, regularization coefficient sequence $\{\tau_t\}_{t=0}^{\infty}$ satisfying (17)

Initialization: $V_0(s) \leftarrow 0,\ \forall s \in \mathcal{S}, t \leftarrow 0$;

repeat

$Q_{t,s}(a,b) \leftarrow r(s,a,b) + \sum_{s'} p(s'|s,a,b)V_t(s')$;

$f_v^{\tau_t}(Q_{t,s}) := \min_{x\in\Delta(\mathcal{A})} \max_{y\in\Delta(\mathcal{B})} \left(x^T Q_{t,s} y + \frac{\tau_t}{2}||x||^2 - \frac{\tau_t}{2}||y||^2\right)$;

$V_{t+1}(s) \leftarrow (1-\alpha)V_t(s) + \alpha f_v^{\tau_t}(Q_{t,s}),\ \forall s \in \mathcal{S}$;

$t \leftarrow t+1$;

until *stopping condition is met*;

Output: $\{V_t(s)\}_{s\in\mathcal{S}}$

Next, we present regularized minimax-V learning algorithm and its pseudocode is given in Algorithm 1. Then, we prove that the generated value sequence converges to the minimax value function V^* when the time-varying regularization coefficient sequence $\{\tau_t\}_{t=0}^{\infty}$ in (16) satisfies given condition, as demonstrated by the following theorem.

Theorem 1. *For $t = 0, 1, \cdots$, let*

$$V_{t+1}(s) = (1-\alpha)V_t(s) + \alpha f_v^{\tau_t}(Q_{t,s})$$
$$Q_{t,s}(a,b) = r(s,a,b) + \sum_{s'} p(s'|s,a,b)V_t(s'),$$

where function $f_v^{\tau_t}$ is defined by (14)*. If the regularization coefficient sequence $\{\tau_t\}_{t=0}^{\infty}$ satisfies $\lim_{t\to 0}\tau_t = 0$ and*

$$\lim_{t\to\infty}\sum_{k=0}^{t}(1-\alpha)^k\tau_{t-k} = 0, \tag{17}$$

then from any initial condition V_0, the function sequence $\{V_t\}_{t=0}^{\infty}$ converges to the minimax value function V^*, i.e., $\lim_{t\to\infty}||V_t - V^*|| = 0$.

The proof of Theorem 1 relies on the following result established in [28].

Lemma 2 (Corollary 5 in [28]). *Consider the process generated by the iteration of Equation* (16) *where $0 < \alpha < 1$. Assume that the process defined by*

$$U_{t+1}(s) = (1-\alpha)U_t(s) + \alpha f_v^{\tau}(Q_s^*)$$

converges to V^*. Assume further that there exist number $0 < \gamma < 1$ and a sequence $\lambda_t \geq 0$ converging to zero such that

$$|f_v^{\tau_t}(Q_s) - f_v^{\tau_t}(Q_s^*)| \leq \gamma||V - V^*|| + \lambda_t$$

holds for all V. Then, the iteration defined by Equation (16) *converge to V^*.*

Then, we can prove Theorem 1.

Proof. By Lemma 2, we only need to verify that the two required assumptions hold here. First, we prove that the process defined by

$$U_{t+1}(s) = (1-\alpha)U_t(s) + \alpha f_v^{\tau_t}(Q_s^*)$$

converges to V^*.

$$\begin{aligned}
||U_{t+1} - V^*|| &= \max_s |U_{t+1}(s) - V^*(s)| \\
&= \max_s |(1-\alpha)(U_t(s) - V^*(s)) + \alpha(f_v^{\tau_t}(Q_s^*) - V^*(s))| \\
&\leq \max_s [(1-\alpha)|U_t(s) - V^*(s)| + \alpha|f_v^{\tau_t}(Q_s^*) - V^*(s)|] \\
&\leq (1-\alpha)\max_s |U_t(s) - V^*(s)| + \alpha \max_s |f_v^{\tau_t}(Q_s^*) - V^*(s)| \\
&= (1-\alpha)||U_t - V^*|| + \alpha \max_s |f_v^{\tau_t}(Q_s^*) - f_v(Q_s^*)| \\
&\leq (1-\alpha)||U_t - V^*|| + \frac{\alpha\tau_t}{2}\max\left\{\frac{|\mathcal{A}|-1}{|\mathcal{A}|}, \frac{|\mathcal{B}|-1}{|\mathcal{B}|}\right\},
\end{aligned}$$

where the last equality holds by Equation (12). Let $C = \max\{\frac{|\mathcal{A}|-1}{|\mathcal{A}|}, \frac{|\mathcal{B}|-1}{|\mathcal{B}|}\}$. Further, for the similar reason, we have that for all $k = 0, 1, \cdots, t$,

$$||U_{k+1} - V^*|| \leq (1-\alpha)||U_k - V^*|| + \frac{\alpha C}{2}\tau_k.$$

Therefore,

$$\begin{aligned}
||U_{t+1} - V^*|| &\leq (1-\alpha)||U_t - V^*|| + \frac{\alpha C}{2}\tau_t \\
&\leq (1-\alpha)((1-\alpha)||U_{t-1} - V^*|| + \frac{\alpha C}{2}\tau_{t-1}) + \frac{\alpha C}{2}\tau_t \\
&= (1-\alpha)^2||U_{t-1} - V^*|| + \frac{\alpha C}{2}((1-\alpha)\tau_{t-1} + \tau_t) \\
&\vdots \\
&\leq (1-\alpha)^{t+1}||U_0 - V^*|| + \frac{\alpha C}{2}\sum_{k=0}^{t}(1-\alpha)^k\tau_{t-k}.
\end{aligned}$$

Since $0 < \alpha < 1$ and $\lim_{t\to\infty}\sum_{k=0}^{t}(1-\alpha)^k\tau_{t-k} = 0$, we have that $lim_{t\to\infty}||U_t - V^*|| = 0$, i.e., the sequence $\{U_t\}_{t=0}^{\infty}$ converges to V^* given any initial condition U_0.

Second, we show that $f_v^{\tau_t}$ is a pseudo-contraction operator, i.e., there exist number $0 < \gamma < 1$ and a sequence $\lambda_t \geq 0$ converging to zero such that for any V_1 and V_2,

$$|f_v^{\tau_t}(Q_{1,s} - f_v^{\tau_t}(Q_{2,s})| \leq \gamma||V_1 - V_2|| + \lambda_t$$

where $Q_{1,s}$ and $Q_{2,s}$ is defined as (9). For all $s \in \mathcal{S}$, we have

$$\begin{aligned}
&|f_v^{\tau_t}(Q_{1,s}) - f_v^{\tau_t}(Q_{2,s})| \\
&= |f_v^{\tau_t}(Q_{1,s}) - f_v(Q_{1,s}) + f_v(Q_{1,s}) - f_v(Q_{2,s}) + f_v(Q_{2,s}) - f_v^{\tau_t}(Q_{2,s})| \\
&\leq |f_v^{\tau_t}(Q_{1,s}) - f_v(Q_{1,s})| + |f_v(Q_{1,s}) - f_v(Q_{2,s})| + |f_v(Q_{2,s}) - f_v^{\tau_t}(Q_{2,s})| \\
&\leq \frac{C}{2}\tau_t + (1-\zeta)||V_1 - V_2|| + \frac{C}{2}\tau_t \\
&= (1-\zeta)||V_1 - V_2|| + C\tau_t,
\end{aligned} \tag{18}$$

where $C = \max\{\frac{|\mathcal{A}|-1}{|\mathcal{A}|}, \frac{|\mathcal{B}|-1}{|\mathcal{B}|}\}$. Since the above relation holds for any two V_1 and V_2, take $V_2 = V^*$ and write V_1 as V, then we obtain that

$$|f_v^{\tau_t}(Q_s) - f_v^{\tau_t}(Q_s^*)| \leq (1-\zeta)||V - V^*|| + \alpha C\tau_t.$$

Since $\lim_{t\to 0} \tau_t = 0$ and $0 < \zeta < 1$, the second required assumption in Lemma 2 holds.

Therefore, by Lemma 2, the sequence $\{V_t\}_{t=0}^{\infty}$ generated by

$$\begin{aligned}
V_{t+1}(s) &= (1-\alpha)V_t(s) + \alpha f_v^{\tau_t}(Q_{t,s}) \\
Q_{t,s}(a,b) &= r(s,a,b) + \sum_{s'} p(s'|s,a,b)V_t(s'),
\end{aligned}$$

converges to V^*.

4 Simulation Results

4.1 Ratio game

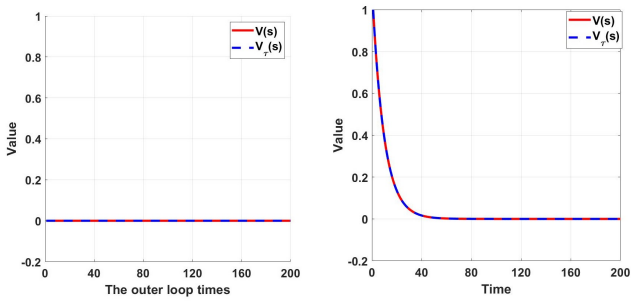
First, we consider ratio game proposed by von Neumann [22]. In ratio game, the state set contains only one state and each player has two actions. Its state transition relation and reward information can be shown by

$$R = \begin{pmatrix} -1 & \epsilon \\ -\epsilon & 0 \end{pmatrix} \quad \text{and} \quad S = \begin{pmatrix} w & w \\ 1 & 1 \end{pmatrix}$$

respectively, where $\epsilon, w \in (0,1)$ with $\epsilon < \frac{1-w}{2w}$. This means that the immediate reward of player Y for selecting actions (a,b) is $R_{a,b}$ and the probability of stopping in each round is $S_{a,b}$. By calculating, given any strategy $x \in \Delta(\mathcal{A})$ and $y \in \Delta(\mathcal{B})$, the value function is

$$V^{x,y} = \frac{x^T R y}{x^T S y}.$$

For the ratio game, there is a unique Nash equilibrium, which is $x^* = y^* = (0,1)$ and thus the minimax value function $V^* = 0$.



(a) The value sequence when initial condition is $V_0 = 0$.

(b) The value sequence when initial condition is $V_0 = 1$.

Fig. 1: The value sequence for ratio game.

We apply regularized minimax-V learning algorithm to solve this game. For comparison, we also consider the performance of the algorithm without introducing the regularization technique. For this algorithm, we call it minimax-V learning algorithm. We set regularization parameter to $\tau_t = (1-\alpha)^t$. For both algorithms, we let them proceed 2000 outer iterations, which are sufficient to ensure convergence.

The induced value sequences with different initial condition are illustrated in Figure 1. Figure 1a shows the value sequence when the initial condition is $V_0 = 0$ while Figure 1b shows the value sequence when the initial condition is $V_0 = 1$. We can see that no matter what the initial condition is, both algorithms converge to the minimax value function. When the initial condition is set to $V_1 = 1$, both algorithms converge in approximately 40 outer iterations, a notably small number compared to the preset 2000 outer iterations.

4.2 Randomly generated Markov game

In this part, we consider a Markov game with randomly generated reward function and state transition function. This Markov game contains 3 states and each player has 3 actions.

For regularized minimax-V learning algorithm, we still set the regularization parameter sequence $\tau_t = (1-\alpha)^t$ and let both algorithms proceed 2000 outer iterations. The value sequences generated by both algorithms are illustrated in Figure 2.

In the figure, the first sub-figure, the second sub-figure and the third sub-figure depict the value sequence of the first, second and third state, respectively. The last sub-figure integrate the first three sub-figures.

This figure shows that the value sequences generated by both learning algorithms gradually converge to the minimax value function and both algorithms only need nearly 200 outer loop rounds to converge.

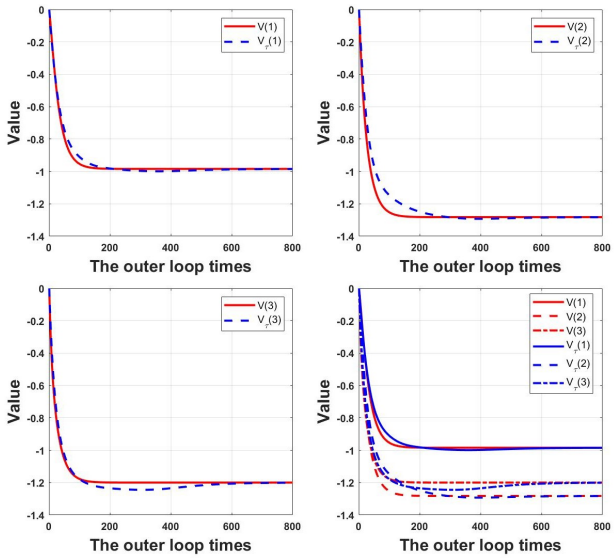


Fig. 2: The value sequence for the randomly generated Markov game: 2000 outer iterations and 5000 inner iterations for both algorithms.

5 Conclusion and Future Work

In this paper, we investigate randomly terminating two-player zero-sum Markov games and propose regularized minimax-V learning algorithm. We introduce the regularization technique to accelerate the process of solving a saddle point problem in the inner loop. The value sequence generated by this algorithm is proven to converge to the minimax value function with appropriate choice of the regularization parameter sequence. Through simulations, we demonstrate that the effectiveness of regularized minimax-V learning algorithm in the ratio game and a randomly generated Markov game.

This paper has some limitations that need to be addressed. Both algorithms require complete information about the reward function and state transition function. In the future, we could consider designing a model-free learning algorithm by replacing the constant learning parameter α with a time-varying learning parameter. Besides, it would be better to provide a theoretical guarantee that regularized minimax-V algorithm converges faster than the minimax-V algorithm. We leave these questions as future work.

Acknowledgments. This work was supported by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant XDA27030201, the Natural Science Foundation of China under Grant T2293770 and the National Key Research and Development Program of China under grant No.2022YFA1004600.

References

1. Agarwal, A., Kakade, S.M., Lee, J.D., Mahajan, G.: Optimality and approximation with policy gradient methods in markov decision processes. In: Conference on Learning Theory. pp. 64–66. PMLR (2020)
2. Bai, Y., Jin, C., Yu, T.: Near-optimal reinforcement learning with self-play. Advances in neural information processing systems **33**, 2159–2170 (2020)
3. Buşoniu, L., Babuška, R., De Schutter, B.: Multi-agent reinforcement learning: An overview. Innovations in multi-agent systems and applications-1 pp. 183–221 (2010)
4. Cai, Y., Oikonomou, A., Zheng, W.: Tight last-iterate convergence of the extragradient and the optimistic gradient descent-ascent algorithm for constrained monotone variational inequalities. arXiv preprint arXiv:2204.09228 (2022)
5. Cen, S., Cheng, C., Chen, Y., Wei, Y., Chi, Y.: Fast global convergence of natural policy gradient methods with entropy regularization. Operations Research **70**(4), 2563–2578 (2022)
6. Cen, S., Chi, Y., Du, S., Xiao, L.: Faster last-iterate convergence of policy optimization in zero-sum markov games. In: International Conference on Learning Representations (ICLR) (2023)
7. Cen, S., Wei, Y., Chi, Y.: Fast policy extragradient methods for competitive games with entropy regularization. Advances in Neural Information Processing Systems **34**, 27952–27964 (2021)
8. Daskalakis, C., Foster, D.J., Golowich, N.: Independent policy gradient methods for competitive reinforcement learning. Advances in neural information processing systems **33**, 5527–5540 (2020)
9. Daskalakis, C., Panageas, I.: Last-iterate convergence: Zero-sum games and constrained min-max optimization. arXiv preprint arXiv:1807.04252 (2018)
10. Fang, X., Zhao, Q., Wang, J., Han, Y., Li, Y.: Multi-agent deep reinforcement learning for distributed energy management and strategy optimization of microgrid market. Sustainable cities and society **74**, 103163 (2021)
11. Filar, J., Vrieze, K.: Competitive Markov decision processes. Springer Science & Business Media (2012)
12. Fudenberg, D., Levine, D.K.: The theory of learning in games, vol. 2. MIT press (1998)
13. Golowich, N., Pattathil, S., Daskalakis, C., Ozdaglar, A.: Last iterate is slower than averaged iterate in smooth convex-concave saddle point problems. In: Conference on Learning Theory. pp. 1758–1784. PMLR (2020)
14. Gorbunov, E., Loizou, N., Gidel, G.: Extragradient method: O (1/k) last-iterate convergence for monotone variational inequalities and connections with cocoercivity. In: International Conference on Artificial Intelligence and Statistics. pp. 366–402. PMLR (2022)
15. Guo, X., Mu, Y., Yang, X.: Periodicity in hedge-myopic system and an asymmetric ne-solving paradigm for two-player zero-sum games. arXiv preprint arXiv:2403.04336 (2024)
16. Jin, C., Liu, Q., Wang, Y., Yu, T.: V-learning–a simple, efficient, decentralized algorithm for multiagent rl. arXiv preprint arXiv:2110.14555 (2021)
17. Kober, J., Bagnell, J.A., Peters, J.: Reinforcement learning in robotics: A survey. The International Journal of Robotics Research **32**(11), 1238–1274 (2013)
18. Konda, V., Tsitsiklis, J.: Actor-critic algorithms. Advances in neural information processing systems **12** (1999)

19. Korpelevich, G.M.: The extragradient method for finding saddle points and other problems. Matecon **12**, 747–756 (1976)
20. Littman, M.L.: Markov games as a framework for multi-agent reinforcement learning. In: Machine learning proceedings 1994, pp. 157–163. Elsevier (1994)
21. Moravčík, M., Schmid, M., Burch, N., Lisỳ, V., Morrill, D., Bard, N., Davis, T., Waugh, K., Johanson, M., Bowling, M.: Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. Science **356**(6337), 508–513 (2017)
22. Neumann, J.v.: A model of general economic equilibrium. The Review of Economic Studies **13**(1), 1–9 (1945)
23. Perolat, J., Munos, R., Lespiau, J.B., Omidshafiei, S., Rowland, M., Ortega, P., Burch, N., Anthony, T., Balduzzi, D., De Vylder, B., et al.: From poincaré recurrence to convergence in imperfect information games: Finding equilibrium via regularization. In: International Conference on Machine Learning. pp. 8525–8535. PMLR (2021)
24. Sayin, M., Zhang, K., Leslie, D., Basar, T., Ozdaglar, A.: Decentralized q-learning in zero-sum markov games. Advances in Neural Information Processing Systems **34**, 18320–18334 (2021)
25. Shapley, L.S.: Stochastic games. Proceedings of the national academy of sciences **39**(10), 1095–1100 (1953)
26. Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of go with deep neural networks and tree search. nature **529**(7587), 484–489 (2016)
27. Syrgkanis, V., Agarwal, A., Luo, H., Schapire, R.E.: Fast convergence of regularized learning in games. Advances in Neural Information Processing Systems **28** (2015)
28. Szepesvári, C., Littman, M.L.: A unified analysis of value-function-based reinforcement-learning algorithms. Neural computation **11**(8), 2017–2060 (1999)
29. Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., et al.: Grandmaster level in starcraft ii using multi-agent reinforcement learning. nature **575**(7782), 350–354 (2019)
30. Wang, X., Sandholm, T.: Reinforcement learning to play an optimal nash equilibrium in team markov games. Advances in neural information processing systems **15** (2002)
31. Wei, C.Y., Lee, C.W., Zhang, M., Luo, H.: Linear last-iterate convergence in constrained saddle-point optimization. arXiv preprint arXiv:2006.09517 (2020)
32. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**, 229–256 (1992)
33. Xie, Q., Chen, Y., Wang, Z., Yang, Z.: Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In: Conference on learning theory. pp. 3674–3682. PMLR (2020)
34. Zhang, K., Yang, Z., Başar, T.: Multi-agent reinforcement learning: A selective overview of theories and algorithms. Handbook of reinforcement learning and control pp. 321–384 (2021)
35. Zhao, Y., Tian, Y., Lee, J., Du, S.: Provably efficient policy optimization for two-player zero-sum markov games. In: International Conference on Artificial Intelligence and Statistics. pp. 2736–2761. PMLR (2022)