I. Proof of Theoretical Results

A. Proof of Proposition I

Theorem 1. (Game Generalized Policy Improvement) Let $\Pi_1, \Pi_2, ...\Pi_n$ be n decision policies and let $\tilde{Q}^{\Pi_1}, \tilde{Q}^{\Pi_2}, ...\tilde{Q}^{\Pi_n}$ be approximates of their respective action value function such that

$$|Q^{\Pi_i}(s,a,b) - \tilde{Q}^{\Pi_i}(s,a,b)| \leq \epsilon$$

for all $s \in S, a \in A, b \in B$ and $i \in \{1, 2, ...n\}$.

Define

$$\pi(s) \in arg \max_a \min_b \min_i \tilde{Q}^{\Pi_i}(s,a,b)$$

We start with the assumptions necessary for this learning algorithm to satisfy the conditions of Theorem 1 in [1] and therefore converge to optimal Q values. The dynamic programming operator defining the optimal Q function is. Proof. Simplifying the notation, let

$$Q_{min}(s,a,b) = min_i Q^{\Pi_i}(s,a,b) \text{ and}$$
$$\tilde{Q}_{min}(s,a,b) = \min_i \tilde{Q}^{\Pi_i}(s,a,b)$$

We start by noting that for any $s \in S$ and any $a \in A$ and any $b \in B$ the following holds:

$$|Q_{min}(s,a,b) - \tilde{Q}_{min}(s,a,b)| = |\min_i Q^{\Pi_i}(s,a,b) -$$
$$\min_i \tilde{Q}^{\Pi_i}(s,a,b)| = \min_i |Q^{\Pi_i}(s,a,b) - \tilde{Q}^{\Pi_i}(s,a,b)| \leq \epsilon$$

This property should remain at a minimum as well as a maximum.

For all $s \in S$ and $a \in A$ and $i \in 1,2$, we have

$$\begin{aligned}
T^{\Pi}\tilde{Q}_{min}(s,a,b) &= r(s,a,b) + \sum_s p(s'|s,a,b)\tilde{Q}_{min}(s', \Pi(s')) \\
&= r(s,a,b) + \sum_s p(s'|s,a,b) \max_a \min_b \tilde{Q}_{min}(s',a,b) \\
&\geq r(s,a,b) + \sum_s p(s'|s,a,b) \max_a \min_b Q_{min}(s',a,b) - \gamma\epsilon \\
&\qquad \text{this is property of Bellman Operator} \\
&\geq r(s,a,b) + \sum_s p(s'|s,a,b) Q_{min}(s', \Pi_i(s')) - \gamma\epsilon \\
&\geq r(s,a,b) + \sum_s p(s'|s,a,b) Q^{\Pi_i}(s', \pi_i(s')) - \gamma\epsilon \\
&= T^{\Pi_i}Q^{\Pi_i}(s,a,b) - \gamma\epsilon \\
&= Q^{\Pi_i}(s,a,b) - \gamma\epsilon
\end{aligned}$$

Since $T^{\Pi}\tilde{Q}_{min}(s,a,b) \geq Q_i^{\Pi}(s,a,b) - \gamma\epsilon$ for any i task, it must be the case that

$$\begin{aligned}
T^{\Pi}\tilde{Q}_{min}(s,a,b) &\geq \min_i Q^{\Pi_i}(s,a) - \gamma\epsilon \\
&= Q_{min}(s,a) - \gamma\epsilon \\
&\geq \tilde{Q}_{min} - \epsilon - \gamma\epsilon
\end{aligned}$$

The Bellman operator in reinforcement learning is said to have two key properties: monotonicity and contraction.

Monotonicity:

Definition: A mapping T is said to be monotonic if, for any two functions V_1 and V_2 such that $V_1 \leq V_2$, pointwise, it follows that $TV_1 \leq TV_2$ pointwise.
In the context of the Bellman operator: If $Q_1 \leq Q_2$ pointwise (meaning $Q_1(s,a) \leq Q_2(s,a)$ for all s and a), then it implies that $TQ_1 \leq TQ_2$. In other words, improving the estimate of the Q-values for state-action pairs will result in an improved estimate after applying the Bellman operator.

Contraction (or contraction mapping) property:

Definition: A mapping T is a contraction if there exists a constant $0 \leq \gamma < 1$ such that, for all functions V_1 and V_2, it follows that $||TV_1 - TV_2|| \leq \gamma||V_1 - V_2||$ where$||.||$ denotes some norm.

In the context of the Bellman operator: If T is a contraction, applying the Bellman operator to two different Q-value functions results in Q-value functions that are closer together. This property is particularly useful in iterative algorithms because it guarantees convergence to a unique fixed point.

In summary, the monotonicity property ensures that improvements in the Q-value estimates lead to improvements after applying the Bellman operator, and the contraction property guarantees the convergence of iterative methods to a unique solution.

These properties are crucial in the analysis of reinforcement learning algorithms, especially those based on iterative methods like value iteration or Q-learning. They provide theoretical guarantees on the convergence of the algorithms and the consistency of the estimated values.

Now we look into the fixed point theorem: Simplifying the Bellman operator under the assumptions of a deterministic policy and a constant function $e(s,a) = 1$ for all s,a.

Starting with the Bellman operator:

$$T^{\pi}Q(s,a) = \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma \sum_{a'} \pi(a'|s')Q(s',a')]. \tag{1}$$

Assumptions: $\pi(a'|s') = 1$ (deterministic policy)
$e(s,a) = 1$ for all s,a

Now, let's apply these assumptions to simplify the Bellman operator:

$$T^{\pi}(\tilde{Q}_{min}(s,a) + ce(s,a)) = \sum_{s'} P(s'|s,a)[R(s,a,s') +$$
$$\gamma \sum_{a'} \pi(a'|s')(\tilde{Q}_{min}(s',a') + ce)].$$

Given the deterministic policy, $\pi(a'|s') = 1$, so the summation over a' simplifies:

$$= \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma(\tilde{Q}_{min}(s',a') + ce)]. \tag{2}$$

Now, since $e(s,a) = 1$ for all s,a, the term $ce(s,a)$ becomes c, and we have:

$$= \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma(\tilde{Q}_{min}(s',a') + c)]. \tag{3}$$

Expanding the sum over s' and using the fact that $P(s'|s,a)$ is a probability distribution and considering the fact that

$\pi(a'|s') = 1$ implies that there is only one relevant a' for each s', we can simplify further:

$$= R(s,a,s') + \gamma(\tilde{Q}_{min}(s',a',) + c)] \tag{4}$$

Finally, the expression becomes:

$$= \tilde{Q}_{min}(s,a) + \gamma c \tag{5}$$

Now, we look into how the property can be used for multi-agent settings, if the policy is deterministic the above property holds:

Now we want to prove that:

First and foremost:

$$\begin{aligned}
Q^{\Pi}(s,a,b) &= \lim_{k\to\infty}(T^{\pi})^{k}\tilde{Q}_{min}(s,a,b) \\
&= lim_{k+1\to\infty}(T^{\pi})^{k+1}T^{\pi}\tilde{Q}_{min}(s,a,b) \\
&\geq lim_{k+1\to\infty}(T^{\pi})^{k}(\tilde{Q}_{min}(s,a,b) - \epsilon(1+\gamma)) \\
&= lim_{k+2\to\infty}(T^{\pi})^{k}T^{\pi}(\tilde{Q}_{min}(s,a,b) - \epsilon(1+\gamma)) \\
&= lim_{k+2\to\infty}(T^{\pi})^{k}(\tilde{Q}_{min}(s,a,b) - \gamma\epsilon(1+\gamma)) \\
&\geq \tilde{Q}_{min} - \epsilon\frac{1+\gamma}{1-\gamma} \text{ : geometric expansion if } \gamma \leq 1 \\
&= Q_{min}(s,a,b) - \epsilon - \epsilon\frac{1+\gamma}{1-\gamma}
\end{aligned}$$

Proof: the result is a direct application of the theorem 1 and Lemma 1. For any j

B. Proof of Lemma 1

Lemma 1.: Let $\delta_{ij} = \max_{s,a,b}|r_i(s,a,b) - r_j(s,a,b)|$ and let Π be an arbitrary policy. Then,

$$|Q_i^{\Pi}(s,a,b) - Q_j^{\Pi}(s,a,b)| \leq \frac{\delta_{ij}}{1-\gamma}$$

Proof. Let us simplify the notation, now let $Q_i^j(s,a,b) = Q_i^{\Pi_j^*}(s,a,b)$. Then,

$$\begin{aligned}
Q_i^i(s,a,b) - Q_i^j(s,a,b) = Q_i^i(s,a,b) - Q_j^j(s,a,b) + \\
Q_j^j(s,a,b) - Q_i^j(s,a,b) \leq \\
|Q_i^i(s,a,b) - Q_j^j(s,a,b)| + |Q_j^j(s,a,b) - Q_i^j(s,a,b)|
\end{aligned}$$

Our strategy will be to bound $|Q_i^i(s,a,b) - Q_j^j(s,a,b)|$ and $|Q_j^j(s,a,b) - Q_i^j(s,a,b)|$ Note that $|Q_i^i(s,a,b) - Q_j^j(s,a,b)|$ is the difference between the value functions of two Markov Games with the same transition function but potentially different rewards.

Define $\Delta_{ij} = \max_{s,a,b}|Q_i^i(s,a,b) - Q_j^j(s,a,b)|$. Then,

$$\begin{aligned}
&|Q_i^i(s,a,b) - Q_j^j(s,a,b)| \\
&= |r_i(s,a,b) + \sum_{s'} p(s' \mid s,a,b) \max_{a'\in A}\min_{b'\in B} Q_i^{\Pi}(s',a',b') \\
&\quad - r_j(s,a,b) - \sum_{s'} p(s' \mid s,a,b) \max_{a'\in A}\min_{b'\in B} Q_j^{\Pi}(s',a',b')| \\
&= |r_i(s,a,b) - r_j(s,a,b) + \sum_{s'} p(s' \mid s,a,b)\big(\max_{a'\in A}\min_{b'\in B} Q_i^{\Pi}(s',a',b') \\
&\quad - \max_{a'\in A}\min_{b'\in B} Q_j^{\Pi}(s',a',b')\big)| \\
&\leq |r_i(s,a,b) - r_j(s,a,b)| + \sum_{s'} p(s' \mid s,a,b)|\max_{a'\in A}\min_{b'\in B} Q_i^{\Pi}(s',a',b') \\
&\quad - \max_{a'\in A}\min_{b'\in B} Q_j^{\Pi}(s',a',b')| \\
&\leq \delta_{ij} + \Delta_{ij}.
\end{aligned}$$

We now turn our attention to $|Q_j^j(s,a,b) - Q_i^j(s,a,b)|$. Following the previous step: define $\Delta'_{ij} = \max_{s,a,b}|Q_j^j(s,a,b) - Q_i^j(s,a,b)|$. Then,

$$\begin{aligned}
&|Q_j^j(s,a,b) - Q_i^j(s,a,b)| \\
&= \left| r_j(s,a,b) + \gamma \sum_{s'} p(s' \mid s,a,b) Q_j^j(s', \Pi_j^*(s')) \right. \\
&\quad \left. - r_i(s,a,b) - \gamma \sum_{s'} p(s' \mid s,a,b) Q_i^j(s', \Pi_j^*(s')) \right| \\
&= \left| r_i(s,a,b) - r_j(s,a,b) \right. \\
&\quad \left. + \gamma \sum_{s'} p(s' \mid s,a,b) \left(Q_j^j(s', \Pi_j^*(s')) - Q_i^j(s', \Pi_j^*(s')) \right) \right| \\
&\leq |r_i(s,a,b) - r_j(s,a,b)| \\
&\quad + \gamma \sum_{s'} p(s' \mid s,a,b) |Q_j^j(s', \Pi_j^*(s')) - Q_i^j(s', \Pi_j^*(s'))| \\
&\leq \delta_{ij} + \Delta'_{ij}.
\end{aligned}$$

Solving for the Δ'_{ij}, you get

$$\Delta'_{ij} \leq \frac{\delta_{ij}}{1-\gamma} \tag{6}$$

Now for the desired result:

$$\begin{aligned}
Q_i^i(s,a,b) - Q_i^j(s,a,b) \leq |Q_i^i(s,a,b) - Q_j^j(s,a,b)| + \\
|Q_j^j(s,a,b) - Q_i^j(s,a,b)| \leq \frac{2\delta_{ij}}{1-\gamma}
\end{aligned}$$

Proposition 1. Let $M_i \in \mathcal{M}^\phi$ and let $Q_i^{\Pi_j^*}$ be the value function of an optimal policy of $M_j \in \mathcal{M}^\phi$ when executed in M_i. Given the set $\{\tilde{Q}_i^{\Pi_1^*}, \tilde{Q}_i^{\Pi_2^*}, ...\tilde{Q}_i^{\Pi_n^*}\}$ such that

$$\begin{aligned} |Q_i^{\Pi_j^*}(s,a,b) - \tilde{Q}_i^{\Pi_j^*}(s,a,b)| \le \epsilon \; for\ all\ s \in S, \\ a \in A\ and\ j \in 1, 2, ...n, \end{aligned}$$

let,

$$\pi(s) \in arg \max_a \min_b \min_j \tilde{Q}_i^{\Pi_j}(s,a,b)$$

Then,

$$\begin{aligned} & Q_i^*(s,a,b) - Q_i^{\Pi}(s,a) \\ \le & \frac{2}{1-\gamma} \max_{s,a,b} |r_i(s,a,b) - r_j(s,a,b)| + \frac{2}{1-\gamma}\epsilon \end{aligned}$$

Proof, The result is a direct application of Theorem 1 and Lemmas 1 and 2. For any $j \in \{1, 2, ...n\}$, we have.

$$\begin{aligned} & Q_i^*(s,a,b) - Q_i^{\Pi}(s,a) \\ & \le Q_i^*(s,a,b) - Q_i^j(s,a) + \frac{2}{1-\gamma}\epsilon \text{ Theorem 1} \\ & \le \frac{2}{1-\gamma}\delta_{ij} + \frac{2}{1-\gamma}\epsilon \text{ Lemma 1} \\ & = \frac{2}{1-\gamma} \max_{s,a,b} |r_i(s,a,b) - r_j(s,a,b)| + \frac{2}{1-\gamma}\epsilon \end{aligned}$$

C. Implementation Details

In this section we describe in detail of the environmental setup and training details of our empirical studies. Pursuer Evader is a standard experiment for zero sum game, we adopted the hyperparameters used in the [2]. We also introduced a challenging task in the Pursuer Evader game which has different initial conditions and more possible goals.

1) Pursuer Evader Qualitative Test: : In Section 5 of the paper we gave an intuitive description of the pursuer evader game used in our experiments. In this section we provide more information on the pursuer evader game and possible output for reach agent. As seen in Fig: 1, there are total of 9 combination of initial position for the agents to move towards the goal and the the exit number infront of each door is the tasks.

Because of the increase in the number of the doors, the new task is described by weights, the length of weights differs from 8 to 10. Hence , like in Case study 1, the task is given by. $[0.7, -1.3, 0.7, 0, 0, 0, 0, 0, 0]$ where the first weight is for manhattan distance between the agents, the next two parameter for the Task 1 (the weight for distance from evader to goal and the weight of terminal reward for evader), after that the subsequent two are for the next task in the increasing order.

2) Algorithms: As mentioned in the Section 4 of the paper, we have both agents updating their TD error at the same time. Figure 2 shows a time scale of how both agents use a one step horizon look out for the other agent's action and choose to maximize based on the current reward.

This has been implemented for all the algorithms where both agents are able to have a one step look ahead into the opponent's policy.

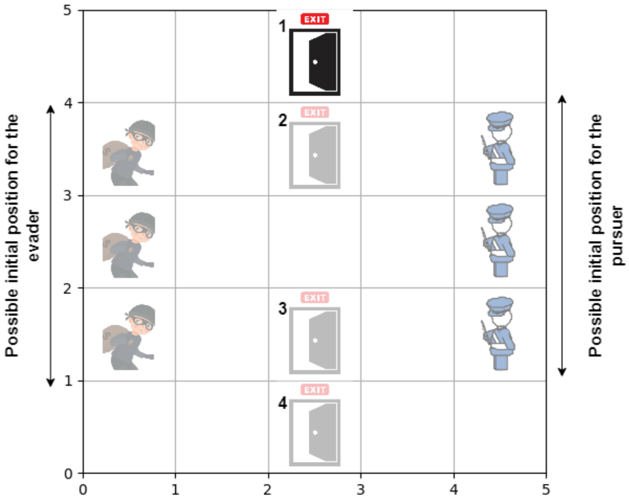
5
4
3
2
1
0
0
1
2
3
4
5
1
2
3
4
EXIT
EXIT
EXIT
EXIT
Possible initial position for the evader
Possible initial position for the pursuer

Fig. 1. All the initial position and goal for the quantitative test.

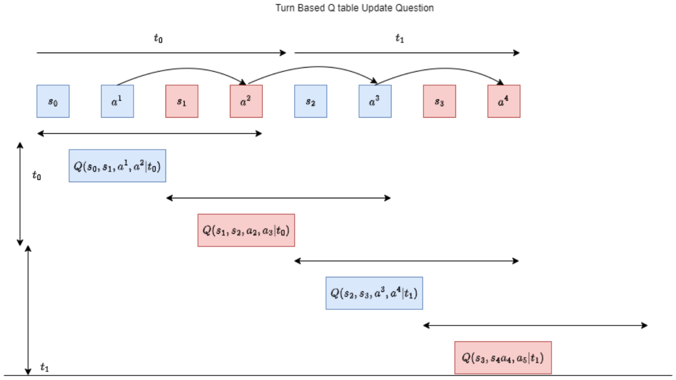
Turn Based Q table Update Question
t_0
t_1
s_0
a^1
s_1
a^2
s_2
a^3
s_3
a^4
t_0
$Q(s_0, s_1, a^1, a^2|t_0)$
$Q(s_1, s_2, a_2, a_3|t_0)$
$Q(s_2, s_3, a^3, a^4|t_1)$
$Q(s_3, s_4a_4, a_5|t_1)$
t_1

Fig. 2. Asynchronous q table update with one step look out horizon for both agents.

References

[1] M. L. Littman and C. Szepesvári, "A generalized reinforcement-learning model: Convergence and applications," in *ICML*, vol. 96, 1996, pp. 310–318.

[2] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver, "Successor features for transfer in reinforcement learning," *Advances in neural information processing systems*, vol. 30, 2017.