We thank all reviewers for their valuable feedback. We would like to first reiterate our main contribution of this paper, and then respond to the individual reviews.

Main Contribution: The main focus of this paper is to gain fundamental understanding of reinforcement learning in the setting of two-player zero-sum games, and especially to investigate the fundamental question "*what is the minimum amount of samples required for provable learning?*". Our results for the first time close this important open problem—i.e. the optimal sample complexity of learning Markov games—in all parameters except episode length. The sample complexities of our newly proposed algorithms also dramatically improve upon the existing ones (see Table 1). We believe our results make a significant contribution to the field of theoretical reinforcement learning.

Reviewer #1. We thank reviewer 1 for the overall positive feedback. Due to the space limit of NeurIPS, we have to cut some explanations/intuitions and defer some to appendix. We will try to provide better explanations and rearrange the materials in the final version.

Reviewer #2. 1. *Memory complexity.* The main focus of this paper is the sample efficiency to learning a Markov game (see the second paragraph of rebuttal). As a tradeoff for obtaining the near-optimal sample complexity, we agree our memory complexity is not completely intractable but still undesirable at the current stage. There are several possible algorithmic ideas that we believe may help improve this memory complexity (for instance, the low-switching idea in "Provably Efficient Q-Learning with Low Switching Cost" paper). Given the amount of results already contained in this paper, we leave memory efficiency for future work.

2. *Why there is no 'natural' way to obtain a policy beside of performing this procedure.* The most standard way we are aware of in the literature of MDP is to directly use the policies the algorithm used in the last episode (or a random episode) during the training process. In online algorithms such as our Nash Q-learning, those policies are guaranteed to perform well only against the Nash equilibrium, and not the best response (which is what we desire). Certified policy is one way we design to fix this problem, but may not be the only way or "necessary". Whether there exists any alternative way remains open, and we believe it is an interesting direction to explore in the future.

3. *Why include algorithm 1.* We include algorithm 1—Optimistic Nash Q-learning—because (1) Nash Q-learning (without optimism) itself is a well-known classical algorithm whose non-asymptotic theoretical guarantee remains absent. Therefore, analyzing a variant of Nash Q-learning may be of independent interest. (2) Algorithm 3 is built upon several algorithmic ideas behind Algorithm 1. They share many common traits, like incremental (model-free) update of value functions, and certified policy. Therefore, algorithm 1 serves as a warmup version to hopefully help the reader understand Algorithm 3, which is more sample efficient but also more involved.

4. *Why optimization problem of equation 9 is always feasible.* Equation 9 is not an optimization problem, and we assume the reviewer actually means equation 8. As mentioned in line 397-399, Nash equilibrium (NE) is a special case of CCE. Since NE always exists, CCE always exists, i.e., the set of linear constraints are always feasible.

5. *Why best response of fixed μ can be non-Markovian?* Only when μ is fixed and *Markovian*, minimizing over ν means solving an MDP. However, when μ is *non-Markovian*, the best response ν can be non-Markovian and dependent on the history. For example, if the max-player follow the strategy that chooses a random action in the first step, and then always pick the same action (same as the first one) in all later steps, the best response would not only depends on the current state, but also depends on the action of the max-player in the first step, thus non-Markovian.

6. *Definition of $\hat{\mu}$ and $\hat{\nu}$.* This is defined in Algorithm 2 for Nash Q-learning and in Algorithm 4 for Nash V-learning. The "hat" version is the actual certified policy (which can be executed as in Algorithm 2 and 4).

Reviewer #3. 1. *A major drawback is that not a single empirical experiment is done.* The main focus of this paper is theoretical, and we believe our theoretical contribution is significant (see the second paragraph of rebuttal). On one hand, we agree experiment/practical performance is important, and it is definitely worth further investigation. On the other hand, given the large amount of pure theoretical ML work without experiments published every year at ICML and NeurIPS, we hope our paper can be evaluated based on its theoretical significance.

2. *Intuitions behind the improvement.* The short answer for the reason behind the improvement is that: Nash Q-learning is an online/incremental update algorithm, which avoids the complicated statistical dependency among the data as in previous algorithm [2], thus shaving off an S factor. This is also briefly explained in line 219-223. Nash V-learning deploys the idea of follow-the-regularized-leader per step, which provides regret guarantee regardless of the number of actions of the opponent, thus reducing the sample complexity from AB to $A + B$. This is also briefly explained in line 239-243. We will add more explanations in the future version.