Learning Mean-Field Games

Anran Hu
UC Berkeley, IEOR
email: anran_hu@berkeley.edu

Joint work with Xin Guo, Renyuan Xu and Junzi Zhang

INFORMS 2019

Outline

Outline

Outline

Motivation: a sequential auction game

Ad auction problem for advertisers:

- Ad auction: a stochastic game on an ad exchange platform among a large number of players (the advertisers)
- Environment: in each round, a web user requests a page, and then a Vickrey-type *second-best-price* auction is run to incentivize advertisers to bid for a slot to display advertisement
- Characteristics:
  - partial information (unknown conversion of clicks)
  - large population

Question: how should one bid in this sequential game with a **large** population of competing bidders and **unknown** distributions of the conversion of clicks/rewards?

Motivation: sequential auction game

Literature

Solution: the $\overbrace{\text{simultaneous learning and decision-making}}^{\text{Reinforcement Learning}}$ problem in a sequential auction with a $\underbrace{\text{large number of homogeneous}}_{\text{Mean-Field Games}}$ bidders.

- **Full model** approach: solve it as an N-player game
  - multi-agent reinforcement learning: computationally intractable
- **Approximation** approaches:
  - independent learners (regarding others as environment) (**IL**)
  - multi-agent reinforcement learning with first-order (expectation) mean-field approximation (**MF-Q**, Yang et al., 2018)
- **Our approach**: Reinforcement Learning (RL) + full distribution Mean-Field Game (MFG) approximation

Outline

Classcial N-player Games

N-player game

$$\begin{aligned} \text{maximize}_{\pi_i} \quad & V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, a_t^i) | \boldsymbol{s}^0 = \boldsymbol{s}\right] \\ \text{subject to} \quad & s_{t+1}^i \sim P^i(\boldsymbol{s}_t, a_t^i) \end{aligned}$$

- N players, state space $\mathcal{S}$, action space $\mathcal{A}$;
- $\boldsymbol{s}_t = (s_t^1, \ldots, s_t^N) \in \mathcal{S}^N$ is the state vector;
- $\boldsymbol{a}_t = (a_t^1, \ldots, a_t^N) \in \mathcal{A}^N$ is the action vector;
- admissible (Markovian) policy $\pi_i : \mathcal{S}^N \to \mathcal{P}(\mathcal{A})$, with $\mathcal{P}(\mathcal{X})$ the space of all probability measures over $\mathcal{X}$;
- r^i is the reward function for player i;
- P^i is the transition dynamics for player i;
- γ is the discount factor;

N-player Games

Definition (N-player game: Nash equilibrium (NE))

NE is a set of strategies such that no agent can benefit from unilaterally deviating from this set of strategies. Formally, $\boldsymbol{\pi}^\star$ is an NE if for all i and $\mathbf{s}$,

$$V^i(\mathbf{s}, \boldsymbol{\pi}^\star) \geq V^i(\mathbf{s}, (\pi_1^\star, \ldots, \pi_i, \ldots, \pi_N^\star))$$

holds for any $\pi_i : \mathcal{S}^N \to \mathcal{P}(\mathcal{A})$.

From N-player Game to MFG

N-player game

$$\begin{aligned} &\text{maximize}_{\pi_i} \quad V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, a_t^i) | \boldsymbol{s}_0 = \boldsymbol{s}\right] \\ &\text{subject to} \quad s_{t+1}^i \sim P^i(\boldsymbol{s}_t, a_t^i). \end{aligned}$$

Assume identical, indistinguishable and interchangeable players.
When the number of players goes to infinity, view the limit of $s_t^{-i} = (s_t^1, \ldots, s_t^{i-1}, s_t^{i+1}, \ldots, s_t^N)$ as population state distribution μ_t.

MFG

$$\begin{aligned} &\text{maximize}_{\pi} \quad V(s, \pi, \{\mu_t\}_{t=0}^{\infty}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, \mu_t) | s_0 = s\right] \\ &\text{subject to} \quad s_{t+1} \sim P(s_t, a_t, \mu_t). \end{aligned}$$

Mean-Field Games (MFG)

MFG

$$\begin{aligned} \text{maximize}_\pi \quad & V(s, \pi, \{\mu_t\}_{t=0}^{\infty}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, \mu_t) | s_0 = s\right] \\ \text{subject to} \quad & s_{t+1} \sim P(s_t, a_t, \mu_t). \end{aligned}$$

- infinite number of homogeneous players, state space $\mathcal{S}$, action space $\mathcal{A}$;
- $s_t \in \mathcal{S}$ and $a_t \in \mathcal{A}$ are the state and action of a representative agent at time t;
- $\mu_t \in \mathcal{P}(\mathcal{S})$ is the population state distribution at time t;
- admissible policy $\pi : \mathcal{S} \times \mathcal{P}(\mathcal{S}) \to \mathcal{P}(\mathcal{A})$;
- r is the reward function, P is the transition dynamics.

Mean-Field Games (MFG)

Definition (Stationary NE for MFGs)

In MFGs, a pair ($\pi^\star$, $\mu^\star$) is called a stationary NE if

1. *(Single agent side) For any policy π and any initial state $s \in \mathcal{S}$, we have*
$$V\left(s, \pi^\star, \{\mu^\star\}_{t=0}^{\infty}\right) \geq V\left(s, \pi, \{\mu^\star\}_{t=0}^{\infty}\right).$$
2. *(Population side) $\mathbb{P}_{s_t} = \mu^\star$ for all $t \geq 0$, where $\{s_t\}_{t=0}^{\infty}$ is the dynamics under control $\pi^\star$ starting from $s_0 \sim \mu^\star$, with $a_t \sim \pi^\star(s_t, \mu^\star)$, $s_{t+1} \sim P(\cdot|s_t, a_t, \mu^\star)$.*

General N-player Games

N-player game

$$\begin{aligned} \text{maximize}_{\pi_i} \quad & V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, a_t^i) | \boldsymbol{s}_0 = \boldsymbol{s}\right] \\ \text{subject to} \quad & s_{t+1}^i \sim P^i(\boldsymbol{s}_t, a_t^i). \end{aligned}$$

General N-player game

$$\begin{aligned} \text{maximize}_{\pi_i} \quad & V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, \boldsymbol{a}_t) | \boldsymbol{s}_0 = \boldsymbol{s}\right] \\ \text{subject to} \quad & s_{t+1}^i \sim P^i(\boldsymbol{s}_t, \boldsymbol{a}_t) \end{aligned}$$

- $\boldsymbol{a_t} = (a_t^1, \cdots, a_t^N)$.

General N-player Games

N-player game

$$\begin{aligned} \text{maximize}_{\pi_i} \quad & V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, a_t^i) | \boldsymbol{s}_0 = \boldsymbol{s}\right] \\ \text{subject to} \quad & s_{t+1}^i \sim P^i(\boldsymbol{s}_t, a_t^i). \end{aligned}$$

General N-player game

$$\begin{aligned} \text{maximize}_{\pi_i} \quad & V^i(\boldsymbol{s}, \boldsymbol{\pi}) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r^i(\boldsymbol{s}_t, \boldsymbol{a}_t) | \boldsymbol{s}_0 = \boldsymbol{s}\right] \\ \text{subject to} \quad & s_{t+1}^i \sim P^i(\boldsymbol{s}_t, \boldsymbol{a}_t) \end{aligned}$$

- $\boldsymbol{a_t} = (a_t^1, \cdots, a_t^N)$.

Generalized Mean-Field Games (GMFG)

MFG

$$\begin{aligned} &\text{maximize}_\pi \quad V(s,\pi,\{\mu_t\}_{t=0}^\infty) := \mathbb{E}\left[\sum_{t=0}^\infty \gamma^t r(s_t,a_t,\mu_t)|s_0=s\right] \\ &\text{subject to} \quad s_{t+1}\sim P(s_t,a_t,\mu_t). \end{aligned}$$

GMFG

$$\begin{aligned} &\text{maximize}_\pi \quad V(s,\pi,\{L_t\}_{t=0}^\infty) := \mathbb{E}\left[\sum_{t=0}^\infty \gamma^t r(s_t,a_t,L_t)|s_0=s\right] \\ &\text{subject to} \quad s_{t+1}\sim P(s_t,a_t,L_t). \end{aligned}$$

- $L_t \in \Delta^{|\mathcal{S}||\mathcal{A}|}$ is the population state-action pair distribution at time t, with state marginal μ_t and action marginal α_t (population action distribution);

Nash Equilibrium in GMFGs

Definition (Stationary NE for GMFGs)

In GMFGs, an agent-population pair $(\pi^\star, L^\star)$ is called a stationary NE if

1. *(Single agent side) For any policy π and any initial state $s \in \mathcal{S}$, we have*
$$V\left(s, \pi^\star, \{L^\star\}_{t=0}^{\infty}\right) \geq V\left(s, \pi, \{L^\star\}_{t=0}^{\infty}\right).$$
2. *(Population side) $\mathbb{P}_{s_t, a_t} = L^\star$ for all $t \geq 0$, where $\{s_t, a_t\}_{t=0}^{\infty}$ is the dynamics under control $\pi^\star$ starting from $s_0 \sim \mu^\star$, with $a_t \sim \pi^\star(s_t, \mu^\star)$, $s_{t+1} \sim P(\cdot | s_t, a_t, L^\star)$, and $\mu^\star$ being the population state marginal of $L^\star$.*

Outline

Fixed point/Three-step approach

- Step 1 (Γ_1): given L, solve the stochastic control problem to get $\pi_L^\star$:

$$\begin{aligned} \text{maximize}_\pi \quad & V(s,\pi,L) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t,L) | s_0 = s\right], \\ \text{subject to} \quad & s_{t+1} \sim P(s_t,a_t,L). \end{aligned}$$

- Step 2 (Γ_2): given $\pi_L^\star$, update from L for one time step to get L' following the dynamics.
- Step 3: Check whether L' matches L, and repeat.

Existence and Uniqueness

Theorem 1 (Guo, Hu, Xu & Zhang, 2019)

For any GMFG, if $\Gamma_2 \circ \Gamma_1$ is contractive, then there exists a unique stationary NE. In addition, the three-step approach converges.

Question: How to solve the GMFG when there is uncertainty in r and P?

Existence and Uniqueness

Theorem 1 (Guo, Hu, Xu & Zhang, 2019)

For any GMFG, if $\Gamma_2 \circ \Gamma_1$ is contractive, then there exists a unique stationary NE. In addition, the three-step approach converges.

Question: How to solve the GMFG when there is uncertainty in r and P?

Outline

Reinforcement learning: Q-learning

- Single agent problem with *unknown* P and r

$$\begin{array}{ll} \text{maximize}_\pi & V(s,\pi) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t) | s_0 = s\right], \\ \text{subject to} & s_{t+1} \sim P(s_t,a_t), \quad a_t \sim \pi(s_t), \quad t \geq 0. \end{array}$$

- Optimal value $V^\star(s) := \max_\pi V(s,\pi)$
- Q-function: $Q^\star(s,a) := \mathbb{E}r(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)} V^\star(s')$
- Bellman equation (for Q-function):

$$Q^\star(s,a) = \mathbb{E}r(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)} \max_{a'} Q^\star(s',a')$$

- Q-learning: stochastic approximation to the Bellman equation:

$$\begin{aligned} &Q^{k+1}(s,a) \\ &\leftarrow (1-\beta_t(s,a))Q^k(s,a) + \beta_t(s,a)\left[r(s,a) + \gamma \max_{a'} Q^k(s',a')\right] \end{aligned}$$

Bridge MFG with RL: Finding NE

Three-step approach revisited:

- Step 1: given L, solve the stochastic control problem to get $\pi_L^\star$:

$$\begin{aligned} \text{maximize}_\pi \quad & V(s,\pi,L) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t,L) | s_0 = s\right], \\ \text{subject to} \quad & s_{t+1} \sim P(s_t,a_t,L). \end{aligned}$$

- Step 2: given $\pi_L^\star$, update from L for one time step to get L' following the dynamics.
- Step 3: Check whether L' matches L.

Bridge MFG with RL: Finding NE

Three-step approach revisited (when P and R are unknown):

- **Step 1**: given L, solve a RL problem with transition dynamics $P_L(s'|s,a) := P(s'|s,a,L)$ and reward $r_L(s,a) := r(s,a,L)$ via Q-learning.
- Step 2: given $\pi_L^\star$, update from L for one time step to get L' following the dynamics.
- Step 3: Check whether L' matches L.

Remark: $\pi_L^\star(s) \in \textbf{argmax}_a\, Q_L^\star(s,a)$. When **argmax** is non-unique, replace it with **argmax-e**, which assigns equal probability to the maximizers.

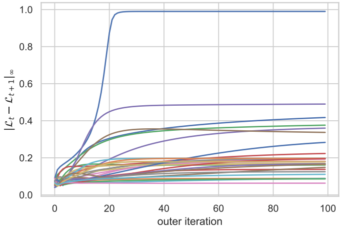
Naive RL Algorithm for GMFG

Algorithm 1 Naive Q-learning for GMFGs

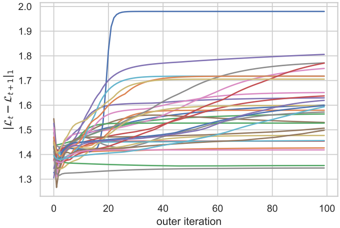
1: **Input**: Initial population state-action pair L_0
2: **for** $k = 0, 1, \cdots$ **do**
3: Perform Q-learning to find the Q-function $Q_k^\star(s,a) = Q_{L_k}^\star(s,a)$ of an MDP with dynamics $P_{L_k}(s'|s,a)$ and reward distributions $R_{L_k}(s,a)$.
4: Solve $\pi_k \in \Pi$ with $\pi_k(s) = \textbf{argmax-e}\,(Q_k^\star(s,\cdot))$.
5: Sample $s \sim \mu_k$, where μ_k is the population state marginal of L_k, and obtain L_{k+1} from $\mathcal{G}(s, \pi_k, L_k)$.
6: **end for**

Failure of the Naive Algorithm

Failure examples:



(a) fluctuation in l_∞.



(b) fluctuation in l_1.

Figure: *Fluctuations of Naive Algorithm (30 sample paths).*

Problems in the Naive Algorithm: Approximation Errors

Algorithm 1 Naive Q-learning for GMFGs

Input: Initial population state-action pair L_0

for $k = 0, 1, \cdots$ **do**

Perform Q-learning to find the Q-function $\overbrace{Q_k^\star(s,a) = Q_{L_k}^\star(s,a)}^{\text{impossible}}$ of an MDP with dynamics $P_{L_k}(s'|s,a)$ and reward distributions $R_{L_k}(s,a)$.

Solve $\pi_k \in \Pi$ with $\pi_k(s) = \overbrace{\textbf{argmax-e}}^{\text{unstable}}(Q_k^\star(s,\cdot))$.

Sample $s \sim \mu_k$, where μ_k is the population state marginal of L_k, and obtain $\underbrace{L_{k+1}}_{\text{unstable}}$ from $\mathcal{G}(s, \pi_k, L_k)$.

end for

Stable Algorithm for GMFG (GMF-Q)

Algorithm 2 Q-learning for GMFGs (GMF-Q)

1: **Input**: Initial L_0, tolerance $\epsilon > 0$.
2: **for** $k = 0, 1, \cdots$ **do**
3: Perform Q-learning for $\boldsymbol{T_k}$ iterations to find the approximate Q-function $\hat{Q}_k^\star(s,a) = \hat{Q}_{L_k}^\star(s,a)$ of an MDP with dynamics $P_{L_k}(s'|s,a)$ and reward distributions $R_{L_k}(s,a)$.
4: Compute $\pi_k \in \Pi$ with $\pi_k(s) = \textbf{softmax}_c(\hat{Q}_k^\star(s,\cdot))$.
5: Sample $s \sim \mu_k$, where μ_k is the population state marginal of L_k, and obtain $\tilde{L}_{k+1}$ from $\mathcal{G}(s, \pi_k, L_k)$.
6: Find $L_{k+1} = \textbf{Proj}_{S_\epsilon}(\tilde{L}_{k+1})$
7: **end for**

Remark. Here S_ϵ is a ϵ-net of L, and $\textbf{softmax}_c(x)_i = \frac{\exp(cx_i)}{\sum_{j=1}^n \exp(cx_j)}$.

Outline

Convergence and Complexity of GMF-Q

Theorem 2 (Guo, Hu, Xu & Zhang, 2019)

Given the same assumptions in the existence and uniqueness theorem, for any specified tolerances $\epsilon,\ \delta > 0$*, set* T_k*,* c *and* S_ϵ *appropriately. Then with probability at least* $1 - 2\delta$*,* $W_1(L_{K_\epsilon}, L^\star) = O(\epsilon)$*, and the total number of iterations* $T = \sum_{k=0}^{K_\epsilon - 1} T_k$ *is bounded by*

$$T = O\left(K_\epsilon^{19/3} \left(\log(K_\epsilon/\delta)\right)^{41/3}\right).$$

Here $K_\epsilon := \left\lceil 2 \max\left\{(\eta\epsilon)^{-1/\eta}, \log_d(\epsilon/\max\{\mathit{diam}(\mathcal{S})\mathit{diam}(\mathcal{A}), 1\}) + 1\right)\right\} \right\rceil$ *is the number of outer iterations.*

Here W_1 is the ℓ_1 Wasserstein distance.

Outline

Repeated Auction Example Revisited

At each round t:

- randomly select $M-1$ players (from N, possibly infinite players) to compete with the representative advertiser
- a_t^M: second best price among the bids from M players
- reward $r_t = \mathbf{I}_{w_t^M=1}\left[(v_t - a_t^M) - (1+\rho)\mathbf{I}_{s_t<a_t^M}(a_t^M - s_t)\right]$
  - v_t: conversion
  - w_t: indicator of winning (bid the highest price)
  - s_t: current budget
  - ρ: penalty of overbidding
- dynamic of the budget:

$$s_{t+1} = \begin{cases} s_t, & w_t \neq 1, \\ s_t - a_t^M, & w_t = 1 \text{ and } a_t^M \leq s_t, \\ 0, & w_t = 1 \text{ and } a_t^M > s_t. \end{cases}$$

- Budget fulfillment: modify the dynamics of s_{t+1} with a non-negative random budget fulfillment $\Delta(s_{t+1})$ after the auction clearing, such that $\hat{s}_{t+1} = s_{t+1} + \Delta(s_{t+1})$.

Performance against full-information

When transition P and reward r are known, replace **Q-learning** with **value iteration** (VI) – **GMF-V**.

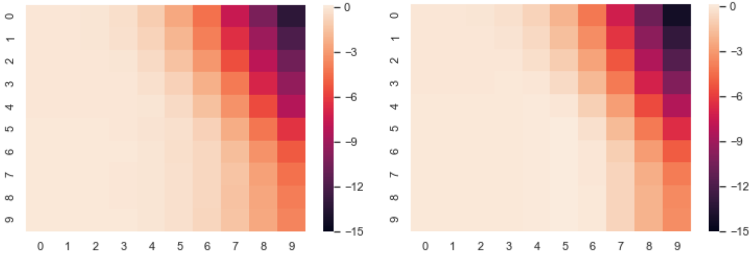
$$Q_L^{k+1}(s,a) \leftarrow \mathbb{E}r(s,a,L) + \gamma \mathbb{E}_{s' \sim P(s,a)} \max_{a'} Q_L^k(s',a'),$$

Table: *Q-table with $T_k^{\text{GMF-V}} = 5000$.*

$T_k^{\text{GMF-Q}}$	1000	3000	5000	10000
ΔQ	0.21263	0.1294	0.10258	0.0989

Here $\Delta Q := \frac{\|Q_{\text{GMF-V}} - Q_{\text{GMF-Q}}\|_2}{\|Q_{\text{GMF-V}}\|_2}$ is the relative L_2 distance between the Q-tables.

Performance against full-information



(a) GMF-Q. (b) GMF-V.

Figure: *Q-tables: GMF-Q vs. GMF-V. 20 outer iterations.*

Conclusion: our algorithm (requiring no specific information on P and R) can learn almost as well as algorithms with full information.

Performance against S.O.T.A.

Performance metric:

$$C(\boldsymbol{\pi}) = \frac{1}{N|\mathcal{S}|^N} \sum_{i=1}^{N} \sum_{\boldsymbol{s} \in \mathcal{S}^N} \frac{\max_{\pi^i} V_i(\boldsymbol{s}, (\boldsymbol{\pi}^{-i}, \pi^i)) - V_i(\boldsymbol{s}, \boldsymbol{\pi})}{|\max_{\pi^i} V_i(\boldsymbol{s}, (\boldsymbol{\pi}^{-i}, \pi^i))| + \epsilon_0}.$$

Here $\epsilon_0 > 0$ is a safeguard, and is taken as 0.1 in the experiments. If $\boldsymbol{\pi}^*$ is an NE, by definition, $C(\boldsymbol{\pi}^*) = 0$ and it is easy to check that $C(\boldsymbol{\pi}) \geq 0$.

Performance against S.O.T.A.

Compare our GMF-Q with IL (independent learners) and MF-Q (N-player game with first-order mean-field approximation, Yang et al., 2018).

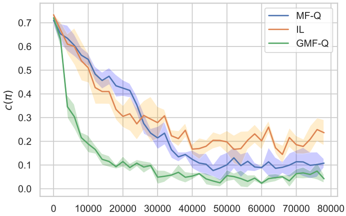


Figure: *Learning accuracy based on* $C(\boldsymbol{\pi})$. $|\mathcal{S}| = |\mathcal{A}| = 10, N = 20$. 90% *confidence interval,* 20 *sample paths.*

Performance against S.O.T.A.

Compare our GMF-Q with IL (independent learners) and MF-Q (N-player game with first-order mean-field approximation, Yang et al., 2018).

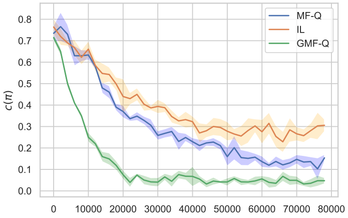


Figure: *Learning accuracy based on* $C(\boldsymbol{\pi})$. $|\mathcal{S}| = |\mathcal{A}| = 20, N = 20$. 90% *confidence interval,* 20 *sample paths.*

Performance against S.O.T.A.

Compare our GMF-Q with IL (independent learners) and MF-Q (N-player game with first-order mean-field approximation, Yang et al., 2018).

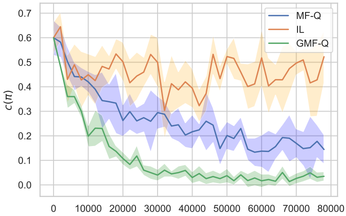


Figure: *Learning accuracy based on* $C(\boldsymbol{\pi})$*.* $|\mathcal{S}| = |\mathcal{A}| = 10, N = 40$*.* 90% *confidence interval,* 20 *sample paths.*

Conclusions

In this work, we

- build a generalized mean-field games framework with learning in a MFG;
- establish the unique existence for the GMFG solution for the discrete time version;
- propose a Q-learning algorithm with convergence and complexity analysis;
- numerical experiments demonstrate superior performance compared to existing RL algorithms.

Thank you!

Reference:

- Guo, X., Hu, A., Xu, R. and Zhang, J. (2019).
  Learning Mean-Field Games.
  arXiv preprint arXiv:1901.09585.

  Shah, D. and Xie, Q. (2018).
  Q-learning with Nearest Neighbors.
  In Advances in Neural Information Processing Systems, pp. 3111-3121.