Reinforcement Learning

Lecture 7: Scalable RL algorithms

Alexandre Proutiere, Sadegh Talebi, Jungseul Ok

KTH, The Royal Institute of Technology

Objectives of the two last lectures

Present algorithmic solutions towards the design of scalable RL algorithms (handling problems with large state, action spaces).

- State aggregation
- Continuous state space
- Value function approximation

Lecture 7: Outline

1. The problem and brief overview of existing solutions
2. RL algorithms with state aggregation
3. RL algorithms for continuous state-space (smooth transition probabilities and rewards)

Lecture 7: Outline

1. **The problem and brief overview of existing solutions**
2. RL algorithms with state aggregation
3. RL algorithms for continuous state-space (smooth transition probabilities and rewards)

RL limitations

RL algorithms presented so far have little chance to solve real-world problems when the state (or action) space is large.

For ergodic RL problems, regret lower bounds scale as $SA\log(T)$ or $\sqrt{SAT}$. Papers on RL algorithms with theoretical guarantees only present simplistic numerical experiments (a few states).

RL with large state space

- Aggregate similar states into a single *meta-state*, and run RL algorithms on meta-state only with the hope of getting a regret with guarantees obtained replacing S by the number of meta-states
- Exploit the smoothness the transition probabilities and the reward function (as a function of states and actions) to accelerate learning (similar to optimal algorithms for bandits with structure)
- Decrease the dimension of the learning problem by *projecting* the value function (or other functions of interest) on a space of small dimension (e.g. deep RL) – covered in Lecture 8
- ...

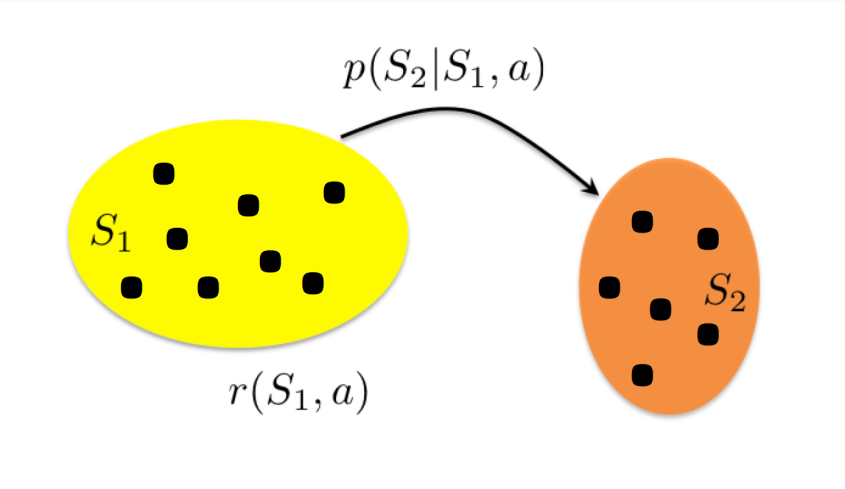
Lecture 7: Outline

1. The problem and brief overview of existing solutions
2. **RL algorithms with state aggregation**
3. RL algorithms for continuous state-space (smooth transition probabilities and rewards)

State aggregation: principle

We restrict our attention to ergodic RL problems.
Imagine we can find a partition $S_1, S_2, \ldots, S_\ell$ of the set of states such that:

$$\forall i, j, \forall s \in S_i, s' \in S_j, \forall a, \quad p(s'|s,a) \approx p(j|i,a), r(s,a) \approx p(i,a).$$



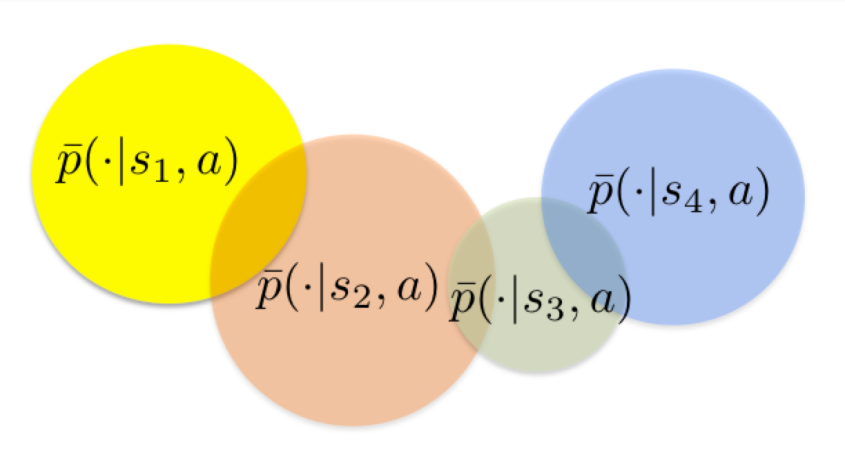
Then, run UCRL2 on the ℓ meta-states would yield a regret no greater than $O(\ell\sqrt{AT})$ ... if identifying the state clusters is regret-free.

Adaptive aggregation for RL[1]

- Bounded Parameter MDP M. Defined by ranges for the MDP parameters:

$$\bar{r}(s,a) = \{r \in [0,1] : |r_c(s,a) - r| \leq d_r(s,a)\}$$
$$\bar{p}(\cdot|s,a) = \{q(\cdot) \in \mathcal{P}(\mathcal{S}) : \|p_c(\cdot|s,a) - q(\cdot)\|_1 \leq d_p(s,a)\}$$

- M is aggregated in $S_1, \ldots, S_\ell$ if for all a, $\cup_{s \in S_i} \bar{r}(s,a)$ and $\cup_{s \in S_i} \bar{p}(\cdot|s,a)$ are connected



[1] R. Ortner, 2013, Annals of Operations Research

Ranges of aggregated BP-MDP

- From a BM-MDP M aggregated in $S_1, \dots, S_\ell$, define the cluster ranges as:

$$\forall j, a, \quad \bar{r}(S_j, a) = \cup_{s \in S_j} \bar{r}(s, a)$$

$$\bar{p}(\cdot|S_j, a) = \{q(\cdot) \in \mathcal{P}(\{1, \dots, \ell\}) : \sum_{S'_j} |p(S'_j|s_0, a) - q(S'_j)| \leq c \max_{s \in S_j} d_p(s, a)\}$$

- UCRL2-like algorithm: in each episode, define the ranges as confidence intervals, aggregate, and find the best MDP within the ranges of the aggregated MDP.

UCAgg

UCAgg is an optimistic algorithm that works in episodes of increasing lengths.

- At the beginning of each episode k,it maintains a set of plausible MDPs $\mathcal{M}_k$ (which contains the true MDP w.h.p.)
- It then aggregates this set of MDPs (state aggregation)
- It finally computes and applies an optimal policy π_k^{agg}, which has the largest gain over all aggregated MDPs

UCAgg

Notations:

- $k \in \mathbb{N}$: index of an episode
- $N_k(s,a)$: total no. visits of pairs (s,a) before episode k
- $\hat{p}_k(\cdot|s,a)$: empirical transition probability of (s,a) made by observations up to episode k
- $\hat{r}_k(s,a)$: empirical reward distribution of (s,a) made by observations up to episode k
- π_k^{agg}: policy followed in episode k
- $\mathcal{M}_k$, $\mathcal{M}_k^{agg}$: BP-MDP for episode k and its aggregation
- $\nu_k(s,a)$: no. of visits of pairs (s,a) seen so far in episode k

UCAgg: Main ingredients

- **BP-MDP: The set of plausible MDPs $\mathcal{M}_k$.** For confidence parameter δ, define

$$\mathcal{M}_k = \Bigg\{ M' = (\mathcal{S}, \mathcal{A}, \tilde{r}, \tilde{p}) : \forall (s,a), |\tilde{r}(s,a) - \hat{r}_k(s,a)| \leq \sqrt{\frac{3.5 \log(2SAt/\delta)}{N_k(s,a)^+}}$$
$$\|\tilde{p}(\cdot|s,a) - \hat{p}_k(\cdot|s,a)\|_1 \leq \sqrt{\frac{14S \log(2At/\delta)}{N_k(s,a)^+}} \Bigg\}$$

- **Aggregation:** Go from $\mathcal{M}_k$ to $\mathcal{M}_k^{agg}$
- **Optimistic gain**: find in $\mathcal{M}_k^{agg}$ the MDP that leads to the highest gain. We need to solve for episode k:

$$\begin{aligned} &\text{maximise over } (M, \pi) \quad g^\pi(M) \\ &\text{subject to } M \in \mathcal{M}_k^{agg} \end{aligned}$$

UCAgg pseudo-code

Algorithm. UCAgg

Input: Initial state s_0, precision δ, $t = 1$

For each episode $k \geq 1$**:**

1. Initialisation. $t_k = t$ (start time of the episode)
   Update $N_k(s,a)$, $\hat{r}_k(s,a)$, and $\hat{p}_k(s,a)$ for all (s,a)
2. Compute BP-MDPs $\mathcal{M}_k$ (using δ) and its aggregate $\mathcal{M}_k^{agg}$
3. Compute the policy
   $\pi_k^{agg} \leftarrow \text{ExtendedValueIteration}(\mathcal{M}_k^{agg}, 1/\sqrt{t_k})$
4. Execute π_k^{agg} and end the episode:
   While $[\nu_k(s_t, \pi_k^{agg}(S_t)) < \max(1, N_k(s_t, \pi_k^{agg}(S_t))]$
   - Play $\pi_k^{agg}(s_t)$, observe the reward and the next state
   - Update $\nu_k(s_t, \pi_k^{agg}(S_t)) \leftarrow \nu_k(s_t, \pi_k^{agg}(S_t)) + 1$ and $t \leftarrow t + 1$

UCAgg: Regret guarantees

Let π =UCAgg Regret up to time T: $\mathcal{R}^\pi(T) = Tg^\star - \sum_{t=1}^{T} r(s_t^\pi, a_t^\pi)$, a random variable capturing the learning cost and the mixing time problems.

Theorem *W.p. at least* $1-\delta$*, the regret of UCAgg satisfies, for any initial state, for any* $T > 1$*,*

$$\mathcal{R}^\pi(T) \leq 49CDS\sqrt{AT\log(\frac{T}{\delta})}.$$

where C *is the maximum aggregation connectivity.*

Remarks

- The previous result does not improve the regret!
- ... essentially because no structural assumptions have been made on the original MDP.
- ... and may be because the exploration process (see length of episodes) does not leverage the aggregation
- What regret gain can we expect with structure?
- UCAgg may not work well for loose structures.

Lecture 7: Outline

1. The problem and brief overview of existing solutions
2. RL algorithms with state aggregation
3. **RL algorithms for continuous state-space (smooth transition probabilities and rewards)**

Smooth MDP

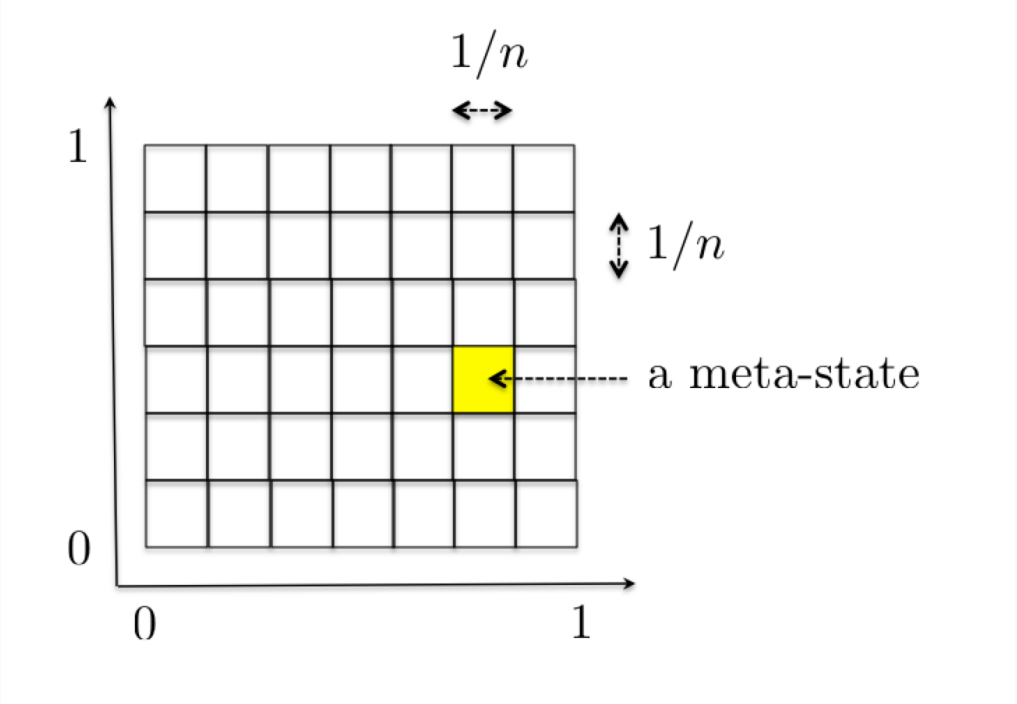
- Ergodic RL problems with continuous state space $\mathcal{S} = [0, 1]^d$ and finite action space $\mathcal{A}$.
- Smoothness of rewards and transition probabilities: for some $\alpha, L > 0$, $\forall s, s', a$,

$$|r(s, a) - r(s', a)| \leq L|s - s'|^\alpha$$
$$\|p(\cdot|s, a) - p(\cdot|s', a)\|_1 \leq L|s - s'|^\alpha$$

- How can this known structure be exploited to reduce regret?

Discretisation

- A possible idea is to discretise the state space and to run UCRL2-like algorithm



Use the smoothness to bound the impact of discretisation on regret.

UCCRL[2]

UCCRL is an optimistic algorithm that works in episodes of increasing lengths. $d = 1$ to simplify the presentation.

- The state space is discretised in $\{I_1, \ldots, I_n\}$. The algorithm runs UCRL2 on these aggregated states.
- At the beginning of each episode k, it maintains a set of plausible aggregated MDPs $\mathcal{M}_k$ (which contains the true MDP w.h.p.)
- It then computes and applies an optimal policy π_k^{agg}, which has the largest gain over all aggregated MDPs

[2] R. Ortner, D. Ryabko, "Online regret bounds for undiscounted continuous reinforcement learning", NIPS 2012

UCCCRL

Notations:

- $k \in \mathbb{N}$: index of an episode
- $N_k(I_j, a)$: total no. visits of pairs (I_j, a) before episode k
- $\hat{p}_k(\cdot|I_j, a)$: empirical transition probability of (I_j, a) made by observations up to episode k
- $\hat{r}_k(I_j, a)$: empirical reward distribution of (I_j, a) made by observations up to episode k
- π_k^{agg}: policy followed in episode k
- $\mathcal{M}_k$: set of plausible MDPs for episode k
- $\nu_k(I_j, a)$: no. of visits of pairs (I_k, a) seen so far in episode k

UCCRL: Main ingredients

- **The set of plausible MDPs $\mathcal{M}_k$.** For confidence parameter δ, define

$$\mathcal{M}_k = \Bigg\{ M' = (\{I_1, \dots, I_n\}, \mathcal{A}, \tilde{r}, \tilde{p}) : \forall (I_j, a),$$

$$|\tilde{r}(I_j, a) - \hat{r}_k(I_j, a)| \leq L n^{-\alpha} + \sqrt{\frac{3.5 \log(2nAt/\delta)}{N_k(I_j, a)^+}}$$

$$\|\tilde{p}(\cdot|I_j, a) - \hat{p}_k(\cdot|I_j, a)\|_1 \leq L n^{-\alpha} + \sqrt{\frac{56n \log(2At/\delta)}{N_k(I_j, a)^+}} \Bigg\}$$

- **Optimistic gain**: find in $\mathcal{M}_k$ the MDP that leads to the highest gain. We need to solve for episode k:

$$\begin{aligned} &\text{maximise over } (M, \pi) \quad g^\pi(M) \\ &\text{subject to } M \in \mathcal{M}_k \end{aligned}$$

UCCRL pseudo-code

Algorithm. UCCRL

Input: $\mathcal{S} = [0,1]$, Initial state s_0, precision δ, $t = 1$, $I_j = (\frac{j-1}{n}, \frac{j}{n}]$, $j = 1, \dots, n$

For each episode $k \geq 1$**:**

1. Initialisation. $t_k = t$ (start time of the episode)
   Update $N_k(I_j, a)$, $\hat{r}_k(I_j, a)$, and $\hat{p}_k(I_j, a)$ for all (I_j, a)
2. Compute $\mathcal{M}_k$ (using δ)
3. Compute the policy
   $\pi_k^{agg} \leftarrow \text{ExtendedValueIteration}(\mathcal{M}_k, 1/\sqrt{t_k})$
4. Execute π_k^{agg} and end the episode:
   While $[\nu_k(I_{i_t}, \pi_k^{agg}(I_{i_t})) < \max(1, N_k(I_{i_t}, \pi_k^{agg}(I_{i_t}))]$
   - Play $\pi_k^{agg}(I_{i_t})$, observe the reward and the next state
   - Update $\nu_k(I_{i_t}, \pi_k^{agg}(I_{i_t})) \leftarrow \nu_k(I_{i_t}, \pi_k^{agg}(I_{i_t})) + 1$ and $t \leftarrow t + 1$

UCCRL: Regret guarantees

Let π =UCCRL Regret up to time T: $\mathcal{R}^\pi(T) = Tg^\star - \sum_{t=1}^{T} r(s_t^\pi, a_t^\pi)$, a random variable capturing the learning cost and the mixing time problems.

Theorem *W.p. at least* $1-\delta$, *the regret of UCCRL satisfies, for any initial state, for any* $T > 1$,

$$\mathcal{R}^\pi(T) = O\left(nH\sqrt{AT\log(\frac{T}{\delta})} + HLn^{-\alpha}T \right).$$

where H *is the maximum bias over considered MDPs.*

For $n = T^{\frac{1}{2+2\alpha}}$, we get $\mathcal{R}^\pi(T) = O(HL\sqrt{A\log(\frac{T}{\delta})}T^{\frac{2+\alpha}{2+2\alpha}}$.

UCCRL: Regret guarantees

- UCCRL exploits the structure *locally* only!
- For Lipschitz continuous transition probabilities, the regret scales as $T^{3/4}$! Is it optimal? (In Lipschitz bandit, naive algorithms are order-optimal and have regret scaling as $T^{2/3}$)
- For practically good algorithms, we need to exploit the structure *globally*! Confidence intervals for $p(\cdot|I_j, a)$ should not be solely defined as a function of $N_k(I_j, a)$. See state-fo-the-art algorithms for Lipschitz bandits.